Université de Marne-La-Vallée

Institut Gaspard Monge

THÈSE

Pour obtenir le grade de

Docteur de l'Université de Marne-La-Vallée

Discipline : Informatique/Télécoms

par

Augustin RADU

le 12 mars 2004

Evaluation de la Qualité de Service par l'utilisateur final dans les systèmes mobiles

Directeur de thèse
Mme Geneviève BAUDOIN

Jury : M. J.P. MARCIANO – professeur Université Aix Marseille - rapporteur
Mme Graziella NICULESCU – professeur Université «Politehnica » Bucarest - rapporteur
M. Makram BOUZID – ingénieur MOTOROLA
Mme Odile PICON – professeur UMLV
Mme Geneviève BAUDOIN – professeur ESIEE – directeur de thèse
M. Bruno SALGUES – Directeur d'études INT

N° attribué par la bibliothèque
| 0 | 0 | M | L | V | | | | | |

(c) Université de Marne-la-Vallée, Augustin Radu - 2004
Origine : http://pelleas.univ-mlv.fr - URL : http://pelleas.univ-mlv.fr/document/UMLV-2004-000235-PDF

A mon Grand Père

Remerciements

Je tiens tout d'abord à remercier plus particulièrement Mme Baudoin pour son aide précieuse dans la réalisation et la rédaction de cette thèse.

Un grand merci à Bruno Salgues et tous les membres du Labo Critic : Olivier, Thomas, Laure, pour leur soutien et les nombreuses heures passées ensemble, sans oublier les stagiaires, Cristina, Costin et Marius Preda, pour leur aide technique.

Merci à Makram qui m'a donné des indications très utiles concernant la partie technique de mon travail.

Merci à tous les permanents de l'INT, à Marie Noëlle pour son assistanat tout au long des ces années.

Merci à Marie Jeanne et Véronique pour avoir pris de leur temps pour relire et corriger mes fautes de français.

Pour finir, un remerciement plus personnel à mes chevaux (Goran, Diego et Lentille) qui m'ont certainement aidé à surmonter des moments difficiles.

Résumé

Les utilisateurs de services multimédia veulent aujourd'hui effectuer l'évaluation de la qualité de service. Pour ce faire, ils ont besoin de procédures et d'outils de mesure appropriés. Malheureusement, la qualité de service et les produits de gestion de la qualité de service disponibles actuellement sur le marché, ne correspondent pas à leurs attentes.

L'objectif de notre recherche vise à combler cette lacune. En partant du principe qu'il est possible de trouver un procédé d'évaluation de la qualité de service simple et commun à tous les utilisateurs, que ce processus d'évaluation peut par ailleurs être automatisé, cette thèse propose un nouveau cadre d'évaluation de la qualité de services, adapté aux intérêts et aux besoins des utilisateurs, orienté vers des applications dans des réseaux indépendants, ouverts et extensibles.

Notre approche porte sur:

1. Une procédure d'évaluation de la qualité de service en 5 étapes conforme à la norme ISO 9000 [Réf. ISO9000].

Le but de cette procédure est d'identifier les paramètres de la qualité qui influencent significativement la perception de la qualité de service par l'utilisateur final. Dans une première étape, tous les paramètres de qualité sont identifiés et exprimés en terme de normes ou de caractéristiques internationales [identification]. Ceux qui sont essentiels pour la perception de la qualité par l'utilisateur sont classés [classification]. Parmi eux, on retient ce qui sont mesurables [sélection] et pour lesquels on peut déterminer des valeurs de seuils [quantification] qui seront utilisées comme limites auxquelles seront comparées [évaluation] les valeurs obtenues lors des différentes mesures.

2. Une architecture d'outils d'évaluation de la qualité de service

Pour être réussie et pour fournir des résultats utilisables, l'évaluation de la qualité de service doit être automatisée. Nous proposons donc une architecture appropriée faisant participer des utilisateurs, des méthodes d'évaluation ainsi que les applications et les services à l'essai. L'utilisation des agents intelligents permet d'automatiser l'évaluation de la qualité de service pour

des applications et des services. L'architecture présentée inclut les modules d'agents et les modèles simples de communication et de coordination pour des agents d'évaluation de la qualité de service.

3. Une plate-forme d'expérimentation de l'architecture d'évaluation de la qualité de service

Une infrastructure d'évaluation de la qualité de service est nécessaire pour établir un environnement de confiance dans lequel les utilisateurs peuvent faire l'évaluation de la qualité de service. Sa première conception inclut des tests sur des services comme le web, le mail et le streaming vidéo comme infrastructure préalable à la qualité normalisée et authentifiée par les opérateurs de télécommunications.

Mots Clés : Qualité de Service, Utilisateur Final, Services Multimédia sur les Mobiles, Agents Intelligents

Abstract

The users of multi-media services today want to be able to evaluate quality of service. In order to do so, they need procedures and suitable measurement tools. Unfortunately, the quality of service and the products for managing quality of service currently available on the market do not correspond to their needs and expectations.

The objective of our research is to fill this gap based on the principle that it is possible to find a simple process of evaluation of the quality of service which would be usable by all users, and which could be automatic. This thesis proposes a new framework for quality of service evaluation, adapted to the users' interests and needs, and intended for applications in independent, open and expandable networks.

Our approach relates to:

1.A procedure of evaluation of the quality of service in 5 stages conforms to the standard ISO 9000 [Réf. ISO9000].

The goal of this procedure is to identify the parameters of quality which significantly influence the perception of the quality of service by the end-user. In a first stage, all the parameters of quality are identified and expressed in term of standards or international characteristics [identification].Those which are essential for the perception of quality by the user are classified [rating].Among them, one retains those which are measurable [selection] and for which we can determine threshold values [quantification] which will be used to compare [evaluation] with the values obtained during various measurements.

2.An architecture of a tool for the quality of service evaluation

Quality of Service evaluation has to be automatic to be successful and to provide usable results. An appropriate architecture involving end users, evaluation methods, and the applications and services under test is designed. The usage of intelligent agents is proposed to do automatic Quality of Service evaluation for applications and services. The architecture introduced includes single agent building blocks and communication and co-ordination models for Quality of Service evaluation agents.

3. An experimentation platform for testing the architecture of quality of service evaluation

A Quality of Service evaluation infrastructure is needed to establish a trusted environment in which end users can do Quality of Service evaluation. Its first design includes tests on services like Web access, email and video streaming as the preliminary infrastructure to the standardised and authenticated quality proposed by the operators in telecommunications.

This thesis is about Quality of Service evaluation and it addresses issues of quality of multimedia applications and distributed services. It fulfils end user requirements.

Keywords: Quality of Service, End User, Mobile Multimedia Services, Intelligent Agents

Table des matières

Remerciements *3*

Résumé *4*

Abstract *6*

Table des matières *8*

Liste des tableaux *12*

Liste des figures *14*

Liste des annexes *15*

1 Introduction *16*

1.1 Problématique et motivations de la thèse **17**

1.1.1 Information technique et désorientation de l'utilisateur final 20

1.1.2 Complexité et dynamique dans les environnements distribués 21

1.1.3 Manque d'Outils Pratique Disponible 22

1.1.4 Un modèle de la QoS 22

1.2 Objectifs de cette thèse **25**

1.3 Importance de cette recherche **26**

1.4 Contribution personnelle **27**

1.5 Organisation du document **29**

2 Nécessité d'une nouvelle évaluation de la QoS *31*

2.1 Terminologie **31**

2.2 Les exigences des utilisateurs **35**

2.2.1 Quels sont les différents types de besoins et d'exigences du client? 37

2.2.2 Pourquoi identifier les besoins et attentes des clients? 38

2.2.3 Liste des souhaits d'un utilisateur pour l'évaluation de la Qualité de Service 40

2.3 Les réseaux d'accès mobiles **41**

2.3.1 Les systèmes « 2G » 41

2.3.2 Les systèmes « 2.5G » 44

2.3.3 Le réseau UMTS (Universal Mobile Telecommunication System) 46

2.3.4 La Qualité des Service dans le monde IP (Internet Protocol) 51
2.3.5 La carte SIM et le SIM ToolKIT 54

2.4 Etat de l'art des principaux travaux effectués 57

2.5 Les normes internationales 62

2.5.1 Le rôle des normes dans les environnements distribués 62
2.5.2 Le choix des critères pour une comparaison 63
2.5.3 Les normes ISO 65
2.5.4 Les Normes ITU-T 69
2.5.5 Les Normes ETSI 80
2.5.6 Les Normes IETF 83

2.6 Résumé Du Chapitre 84

3 Un cadre d'évaluation de la Qualité de Service 86

3.1 Le Modèle d'un Mécanisme pour l'Evaluation de la qualité du service 86

3.1.1 L'Utilisateur final 88
3.1.2 L'opérateur de téléphonie mobile 89
3.1.3 Registre 90

3.2 Un procédé en 5 pas pour l'évaluation de la Qualité de Service 91

3.2.1 Etape 1 : Identification de tous les paramètres de la qualité 93
3.2.2 Etape 2 : Le classement 94
3.2.3 Etape 3 : Choix des Mesures 95
3.2.4 Etape 4 : Quantification des Paramètres mesurables 96
3.2.5 Etape 5 : L'évaluation de la qualité de service 97

3.3 Une Architecture d'Evaluation de la Qualité de Service 99

3.3.1 Qu'est ce qu'un Agent ? 99
3.3.2 Caractéristiques des Agents 101
3.3.3 Types d'Agents 102
3.3.4 Fonctionnement de l'Agent Intelligent 104
3.3.5 Une structure d'un système Multi-Agents 106

3.4 Une architecture d'Agent d'Evaluation de la Qualité de Service 107

3.5 Le Langage Java 110

3.5.1 Caractéristiques du Langage Java 110
3.5.2 Mécanisme général du langage Java 110
3.5.3 Java mode d'emploi 111
3.5.4 Le Java 2 Micro Edition (J2ME) 112

3.6 Résumé du chapitre 119

4 Application de la procédure d'évaluation de la qualité de service 120

4.1 Evaluation de la Qualité de Service pour le courrier électronique 120

4.1.1 Identification de tous les paramètres de la qualité 123

4.1.2 Etape 2: Le classement 125

4.1.3 Etape 3: Choix de mesures 126

4.1.4 Etape 4: Quantification des Paramètres mesurables 127

4.1.5 Etape 5: L'évaluation de la qualité de service 128

4.2 Evaluation de la Qualité de Service pour les services en ligne (www) 129

4.2.1 Identification de tous les paramètres de la qualité 132

4.2.2 Etape 2: Le classement 136

4.2.3 Etape 3: Choix de mesures 138

4.2.4 Etape 4: Quantification des Paramètres mesurables 139

4.2.5 Etape 5: L'évaluation de la qualité de service 140

4.3 Evaluation de la Qualité de Service pour le streaming vidéo 141

4.3.1 Le streaming vidéo 141

4.3.2 Techniques de compression video 143

4.3.3 Compression audio-video MPEG-1 146

4.3.4 Identification de tous les paramètres de la qualité 148

4.3.5 Etape 2: Le classement 152

4.3.6 Etape 3: Choix de mesures 153

4.3.7 Etape 4: Quantification des Paramètres mesurables 154

4.3.8 Etape 5: L'évaluation de la qualité de service 155

4.4 Résumé du chapitre 155

5 Un prototype d'implémentation avec des Agents Intelligents pour simuler l'évaluation de la qualité de service 157

5.1 Implémentation des Agents Intelligents pour mettre en oeuvre l'évaluation de la qualité de service pour différents services mobiles 157

5.2 Plateforme JADE-LEAP 160

5.3 Architecture d'un Agent pour évaluation de la QoS pour les services on-line (web) 164

5.3.1 L'Agent Web 164

5.4 Architecture d'un Agent pour l'évaluation de la qualité de service pour les emails 168

5.4.1 Agent Mail 169

5.5 Architecture d'un Agent pour évaluation de QoS pour le streaming vidéo 173

5.5.1 L'Agent Vidéo 173

5.6 Simulation des Agent Web, Mail et Vidéo sur la plateforme JADE-LEAP à l'aide du J2ME Wireless Toolkit 178

5.7 Forme de résultats obtenus 179

5.8 Résumé du chapitre 180

6 Conclusion et travail futur 182

6.1 Résumé de la thèse 182

6.2 Problèmes rencontrés et points non résolus de la thèse 184

6.3 Travail futur 186

6.4 Implications potentielles 187

Annexes 194

Liste des tableaux

Tableau 1-1 Une taxonomie des applications réseau. 24
Tableau 2-1 Définition des indicateurs de la qualité de service. 42
Tableau 2-2 Classes de fiabilité de GPRS 45
Tableau 2-3 Classes de délai de GPRS 45
Tableau 2-4 Evolution de services GSM vers l'UMTS. 49
Tableau 2-5 Le modèle XRM 59
Tableau 2-6 Choix des critères 64
Tableau 2-7 Distinction entre la qualité de service et les performances du réseau 70
Tableau 2-8 Relation qualitative entre les paramètres de performance génériques et les paramètres de QOS possibles pour les services supports 73
Tableau 2-9. Différentes niveaux de QoS liés au point de vue ODP 79
Tableau 2-10 Matrice pour faciliter la prise en compte des exigences des utilisateurs 82
Tableau 2-11 Les normes internationales examinées concernant QoS. 84
Tableau 3-1 Note d'appréciation 98
Tableau 3-2 Une procédure en 5 étapes d'évaluation de la qualité de service 99
Tableau 3-3 Propriétés des Agents 101
Tableau 3-4 Liste de packages CDLC 116
Tableau 3-5 Les packages MIDP 118
Tableau 4-1 Caractéristiques du logiciel ISO 9126 124
Tableau 4-2 Les paramètres essentiels du service mail 125
Tableau 4-3 Paramètres essentiels de la qualité de service à mesurer pour le mail 127
Tableau 4-4 Paramètre valeur pour le mail 128
Tableau 4-5 WWW Service. 134
Tableau 4-6 WWW Protocol 135
Tableau 4-7 WWW Document 135
Tableau 4-8 Description wwwServiceEntry 137
Tableau 4-9 Description wwwSummaryEntry 138
Tableau 4-10 Choix de paramètres MIB par rapport aux paramètres QoS 139
Tableau 4-11 Paramètres valeurs pour les services en ligne WWW 140
Tableau 4-12 Paramètres essentiels et non essentiels audio pour une séquence vidéo MPEG 1 149
Tableau 4-13 Paramètres essentiels et non essentiels image pour une séquence vidéo MPEG 1 150
Tableau 4-14 Paramètres essentiels et mesurables pour l'accès à une séquence audio/vidéo 151
Tableau 4-15 Paramètres essentiels de la qualité de service pour le streaming audio/vidéo 152
Tableau 4-16 Paramètres essentiels mesurables de la qualité de service pour le streaming audio/vidéo 153
Tableau 4-17 Paramètres valeurs pour le service de streaming audio/vidéo 154

Tableau 5-1 Paramètres valeurs pour les services on-line WWW _____ *165*
Tableau 5-2 Paramètres valeurs pour le service de streaming vidéo _____ *173*
Tableau 6-1 Les normes internationales examinées et complétées concernant QoS _____ *183*

Liste des figures

Figure 1-1 Environnement Distribué avec les différentes couches de la QoS *23*
Figure 2-1 Les Dépendances de la Qualité [Bissel00a] *36*
Figure 2-2 Différents types de Qualité [AFNOR] *38*
Figure 2-3 Taux de couverture réalisée et obligations des opérateurs mobiles. *43*
Figure 2-4 Différentiation du trafic sur un réseau IP *52*
Figure 2-5 Modèle Intserv pour les réseaux IP *53*
Figure 2-6 Modèle Diffserv pour les réseaux IP *54*
Figure 2-7 Méthode systématique d'identification et d'organisation paramètres de performance du réseau *71*
Figure 3-1 Modèle d'un Mécanisme pour l'évaluation de la qualité du service sur les mobiles. *87*
Figure 3-2 Procédé d'évaluation de la qualité service. *92*
Figure 3-3 Etape 1 : Identification de tous les paramètres de la qualité *93*
Figure 3-4 Etape 2: Le classement *95*
Figure 3-5 Etape 3 : Choix des Mesures *96*
Figure 3-6 Etape 4: Quantification dans des paires Paramètres Valeurs *97*
Figure 3-7 Etape 5 : Evaluation de la qualité de service *98*
Figure 3-8 Mécanisme de l'intelligence chez un agent *105*
Figure 3-9 Structure d'un système multi-agents *106*
Figure 3-10 Une architecture d'Agent *108*
Figure 3-11 Architecture Plateforme Java 2 Micro Edition *113*
Figure 3-12 Différentes couches de J2ME *113*
Figure 3-13 Configuration, Profile J2ME *116*
Figure 3-14 Wireless Sim Toolkit *118*
Figure 3-15 Le DefaultGreyPhone *119*
Figure 4-1 Acheminement du courrier électronique *121*
Figure 4-2 Relation entre les images de type I et P *145*
Figure 4-3 Relation entre les images de type I, P et B *145*
Figure 5-1 Lancement plateforme Jade *161*
Figure 5-2 Architecture de référence pour les Plateformes FIPA *161*
Figure 5-3 Couche fonctionnelle de la plateforme JADE-LEAP *164*
Figure 5-4 Description du fonctionnement de l'Agent Web » *166*
Figure 5-5 Structure interne de l'Agent Manager *167*
*Figure 5-6 Architecture de l'**Agent Web*** *168*
Figure 5-8 Description du fonctionnement de l'Agent Web-Mail » *171*

Liste des annexes

Annexe 1 Résultats de l'évaluation du mail *1*
Annexe 2 Résultats de l'évaluation de l'accès web *2*
Annexe 3 Résultats de l'évaluation du streaming audio-vidéo *3*
Annexe 4 Publications *4*
Annexe 5 Classes JAVA des Agents *5*

1 Introduction

La Première Génération Analogique de communications mobiles a laissé place à la Seconde Génération Numérique actuelle. Dans les 5 ans à venir la Troisième Génération devrait être installée et elle permettra l'utilisation du mobile multimédia.

Les exemples typiques de systèmes de la Première Génération sont les réseaux comme C-netz en Allemagne, RadioCom2000 en France et AMPS aux Etats Unis. Les mobiles analogiques on été remplacés du début des années 90 par le système européen de Seconde Génération appelé le GSM (Global System for Mobile Communication) qui a été un grand succès international. Le système GSM est opérationnel dans 3 bandes de fréquences GSM 900 et DCS 1800 MHz en Europe et GSM 1900 MHz aux Etats Unis.

L'évolution actuelle du GSM correspond au GPRS (General Packet Radio Service) qui par son utilisation de la communication en mode paquets et l'augmentation des débits, ouvre les portes vers des applications de mobiles multimédia et permet une transition vers l'UMTS (Universal Mobile Telecommunication System). Une présentation plus détaillée de ces réseaux mobiles sera faite dans le chapitre suivant.

La nécessité d'améliorer le service rendu au niveau des réseaux mobiles s'est largement manifestée, dans le secteur privé comme dans le secteur public. La notion de qualité de service est un indicateur qui nous est de plus en plus familier, mais qui la plupart du temps reste flou dans les esprits car le terme de service recouvre des réalités variées. Nous employons le mot service pour désigner l'expérience des relations vécues à l'occasion de l'achat d'un produit, mais il ne faut pas limiter cette appréciation au moment de l'acquisition de l'objet. Prenons le cas simple de l'achat d'un téléphone portable. Si le besoin du moment (l'acquisition d'un téléphone mobile) est rempli, le client n'a aucune garantie concernant les services qui lui sont proposés ensuite. Quand nous parlons de services, dans notre cas, nous faisons référence à des services proposés par l'opérateur de télécommunications ou tout simplement les fonctionnalités du portable lui-même. La notion de service s'inscrit dans la durée et nécessite une garantie.

Si autrefois les artisans réalisaient leurs produits selon les règles de l'art souvent inconnues par les clients, les opérateurs de télécommunication ne peuvent plus se permettre la même chose. Les performances de leurs réseaux sont généralement connues par les utilisateurs potentiels. En même

temps elles offrent une garantie des services proposés et donc un moyen de fidélisation de leurs clients.

Le domaine actuel de la qualité de service dans le monde des télécommunications est extrêmement vaste puisque les opérateurs et les fournisseurs de service utilisent des technologies différentes et s'adressent à une clientèle très variée. Pour limiter ce domaine et pour faciliter la compréhension et la mise un place d'une approche simple concernant la QoS, nous nous appuyons sur les principes suivants :

- la qualité du service ne concerne que l'ensemble des propriétés, des caractéristiques et des paramètres qui peuvent être choisis, mesurés et comparés à des valeurs limites.
- l'évaluation de la qualité du service peut être réduite à quelques caractéristiques essentielles de qualité. Il n'est pas nécessaire de définir et mesurer chaque propriété des dispositifs du service.
- il est important de définir un ensemble commun d'outils pour fournir des résultats comparables non seulement pour faire face à la concurrence, mais aussi pour fidéliser les utilisateurs finaux.

L'évolution des services, la complexité de leur mise en ouvre et le retard qui a été pris dans leur développement nous ont conduit à la problématique de cette thèse.

1.1 Problématique et motivations de la thèse

Dans le cas des réseaux de seconde ou troisième génération, la problématique de la qualité de service est complexe car nous sommes confrontés d'une part à un environnement multi-services et multimédia avec différentes exigences de bon fonctionnement et d'autre part à deux modes de transport de l'information : circuit ou paquet. Les applications sont de natures variées telles que la voix, la vidéoconférence, la messagerie électronique, le commerce électronique, la vidéo, le transfert de fichiers, les SMS, le chargement d'images….Certaines de ces applications ont des contraintes de transmission en temps réel et d'autres non, certaines sont tolérantes aux erreurs de transmission et d'autres non. Elles demandent des débits plus ou moins élevés.

A telecommunications service is a particular capability to communicate with other parties by transmitting and receiving information in a way that is fully specified with respect to: how the user

initiates a transaction; the mode in which the information is exchanged; how the information is formatted for transmission; how end-to-end exchanges of the information are effected; and how the transactions completed are billed and paid for. The important distinction in concept between the service and the systems or capabilities that deliver it is that users, as opposed to providers of the service, experience and care about only those characteristics of the service that are manifested when they try to effect the end-to-end communications transactions.

Un service de télécommunication se caractérise par l'aptitude à communiquer avec d'autres entités par la transmission et la réception des informations qui sont définies au préalable en ce qui concerne : comment un utilisateur lance une communication, le moyen par lequel l'information est échangée, comment l'information est stockée pour être transmise, comment les échanges d'informations de bout en bout sont effectués et comment les systèmes de facturation sont mis en place. La différence entre les utilisateurs et les fournisseurs de services est que les utilisateurs ne s'intéressent qu'aux caractéristiques du service qui se manifestent lorsqu'ils essayent d'effectuer une communication de bout en bout. [Réf. Hardy] .

Nous pouvons établir une analogie entre la gestion des services au sein d'un réseau et celui de la gestion d'un réseau routier ; chaque service ou application représente un type de véhicule (un vélo, une voiture, un camion, une moto…), le point de départ d'un véhicule est le point d'accès à l'application (serveur Web …), le point d'arrivée est l'utilisation du service, respectivement de l'application. De plus chaque véhicule dispose d'un temps donné pour aller de son point de départ à son point d'arrivée.

En l'absence de règles de circulation, les voies de communication seraient rapidement bloquées , les artères engorgées, et le délai de déplacement d'un point à un autre excessif.

L'existence de règles de priorité et de mécanisme de régulation (feux, panneaux, voies dédiées aux bus et camions) associés à un dimensionnement cohérent des voies de circulation permet de réguler le trafic dans des conditions normales d'utilisation et d'assurer une bonne qualité de service.

Il est donc fondamental d'établir, de façon analogue à la gestion du trafic dans les réseaux routiers, une architecture cohérente, un ensemble de règles et des mécanismes au niveau du réseau, de l'administration du réseau et des services. Si nous disposons d'outils d'analyse de chacun des segments du réseau (segment radio, segment filaire, Internet, etc.), nous ne possédons pas de modèle qui place l'utilisateur final au centre des préoccupations de la qualité de service.

Cette thèse propose un modèle pour l'évaluation de la qualité du service (QoS) pour des applications multimédia qui s'appuie sur la norme internationale ISO 9000 [Réf. ISO9000] et les recommendations ITU-T [E.800].

Quality of Service evaluation is the systematic examination of the extent to which an entity (e.g., a process, a product, a service, an application) is capable of fulfilling specified requirements.

L'évaluation de la qualité de service désigne l'ensemble des caractéristiques d'une entité (un processus, un produit ou une application), relatif à sa capacité à satisfaire les besoins spécifiés [Réf. IS08402].

L'évaluation de la qualité de service doit être facilitée pour que les utilisateurs soient satisfaits. Nous proposons que dans leur rôle de consommateurs de service, les utilisateurs puissent réellement procéder à l'évaluation de QoS, à condition qu'ils disposent d'un certain appui et des conseils techniques du Service Clientèle de l'opérateur, d'un ensemble approprié d'outils d'évaluation et d'un environnement digne de confiance.

Afin de définir et de gérer les niveaux de service auxquels leurs clients peuvent prétendre, les opérateurs mobiles doivent définir des indicateurs mesurables par rapport aux services proposés. Ces indicateurs seront convertis en garanties de service contractuelles en proposant des SLA (Service Level Agreement, accord de niveau de service) aux clients. Ceux-ci, en effectuant l'évaluation de la qualité de service avec leur mobile pourrons vérifier si l'accord de niveau de service signé avec leur opérateur ou leur fournisseur de service est effectivement respecté.

L'utilisateur veut des données factuelles. Il veut pouvoir évaluer la qualité de service en temps réel, et non de façon théorique. Il veut pouvoir comparer la qualité des différentes offres de service et les garanties de service des candidats fournisseurs de service. En l'absence d'une compréhension claire, fondée sur des faits dignes de confiance, il risque de payer trop ou d'obtenir trop peu de son fournisseur de service. En outre, il veut connaître les tendances qui permettront de mettre en place des étapes pro-actives pour la gestion de la qualité de service et non plus de simples mesures réactives.

L'ITU-T [Réf. ITU-T00], autrefois appelé CCITT, qui est l'organisme responsable de la normalisation des télécommunications sur le plan mondial, définit la QoS de la manière suivante :

La qualité de services représente l'effet global procuré par la qualité de fonctionnement d'un service qui détermine le degré de satisfaction de l'usager.[Réf. I.350]

Cette définition est peu précise puisque qu'elle utilise deux termes « effet » et « satisfaction », qui ont plusieurs significations possibles. Pour que l'évaluation de la qualité de service soit réussie, elle doit être menée avec la participation des utilisateurs qui sont la cible principale.

ITU-T précise dans I.350 une suggestion qui vise à aider les utilisateurs en déclarant que la qualité du service est mieux exprimée par les paramètres qui :

- se concentrent sur la perception par l'usager, plutôt que leurs causes dans le réseau ;
- ne dépendent pas dans leur définition, des performances techniques du réseau ;
- tiennent compte des aspects du service du point de vue de l'utilisateur qui peuvent être objectivement mesurés au point d'accès au service;
- peuvent être assurés à un utilisateur au point d'accès au service par les fournisseurs de service;
- sont décrits en termes indépendants de réseau et créent un langage commun compréhensible par l'utilisateur et le fournisseur de service.

I.350 identifie également l'intérêt de l'utilisateur par rapport à la QoS:

Un usager normal ne se préoccupe pas de la manière dont est fourni un service donné ou des aspects divers de la conception interne d'un réseau. Toutefois, il s'intéresse à la comparaison entre un service et un autre du point de vue de certains problèmes de performance universels, orientés usager qui s'appliquent à n'importe quel service de bout en bout.

Cette thèse suit cette suggestion, partage le point de vue mentionné, et dans un certain sens est de ce fait une prolongation de I.350.

1.1.1 Information technique et désorientation de l'utilisateur final

Nous ne pouvons pas supposer que les utilisateurs de services aient une connaissance précise de l'ensemble et de la valeur réelle des paramètres de la QoS. Normalement les utilisateurs ont une idée au sujet des propriétés importantes ("doivent avoir") et facultatives ("peuvent avoir") d'un service ou d'une application. Cependant, ils ne connaissent qu'une partie des caractéristiques de la qualité de service qui sont liées à la mise en place du service et qui ne sont pas forcement les plus

significatives. En même temps, ils sont dans l'incapacité de trouver des réponses par rapport à la qualité de service, à l'exception des données techniques qui sont souvent assez difficiles à comprendre.

Le service clientèle où le service après-vente de chaque opérateur se trouve souvent dans l'incapacité de répondre aux besoins de leurs clients en terme de QoS.

1.1.2 Complexité et dynamique dans les environnements distribués

Les systèmes que nous utilisons actuellement (station UNIX, PC, terminaux mobiles, etc.), fonctionnent aujourd'hui rarement de façon totalement indépendante. Nous employons le terme d'environnements distribués pour des systèmes connectés à un **réseau** informatique (LocalTalk, Ethernet, FDDI...). Ils utilisent des **ressources** externes (impression, messagerie, stockage de masse, puissance de calcul, etc...) en faisant appel, de façon explicite ou transparente, à des **services** informatiques dits **distribués**. Basés sur des **protocoles** standardisés, ces services sont interopérables entre différentes plate-formes matérielles (milieu **hétérogène** multi-constructeurs) et sont le plus souvent basés sur le modèle **client/serveur** : l'équipement qui fait appel à un service (généralement le poste de travail ou le mobile) étant le **client**, celui qui fournit ce service étant le **serveur**.

L'évaluation de la qualité de service dans les environnements distribués est très complexe.En effet, un service n'est pas nécessairement associé à une machine unique, mais plutôt à un ensemble de composants logiciels répartis sur plusieurs ordinateurs, eux mêmes interconnectés par des réseaux utilisant le plus fréquemment des protocoles IP, Frame Relay ou ATM. Les mesures de la performance et de la qualité de service s'en trouvent sensiblement modifiées, puisqu'elles font intervenir une multitude de paramètres indépendants, provenant non seulement des composants applicatifs formant le service, mais également des systèmes d'exploitation, des équipements réseau ainsi que des liaisons les interconnectant (LAN[1], WAN[2]) [Réf. FOUQUET].

Les causes de l'échec ou de la dégradation de la qualité de service peuvent être diverses. Les symptômes sont parfois détectés plus rapidement par les utilisateurs que les causes perçues par les fournisseurs de service. La question est donc : Que peuvent faire les utilisateurs dans ce cas ? Comment faire ou à qui s'adresser ?

---

[1] LAN – Acronyme de Local Area Network
[2] WAN – Acronyme de Wireless Area Network

1.1.3 Manque d'Outils Pratique Disponible

D'un côté, les utilisateurs ne veulent pas se fier aux résultats des mesures effectuées par le fournisseur de service lui-même. Ils veulent une information impartiale. D'un autre côté, ils ne disposent pas à l'heure actuelle d'outils leur permettant d'évaluer eux-mêmes la qualité de service. Le « *ping*[3] » ne suffit pas! Enfin, ils ont également besoin d'un outil de base reflétant et identifiant les caractéristiques clé de la QoS. Le problème reste posé : la seule manière fiable pour que les utilisateurs évaluent la qualité de service est de la comprendre et de l'évaluer systématiquement eux-mêmes. Mais dans ce cas, la méthodologie doit être simple.

1.1.4 Un modèle de la QoS

Les modèles traditionnels d'évaluation de la QoS de bout en bout se résument à des indicateurs techniques de chaque composant matériel ou logiciel qui pour des raisons économiques et technologiques proviennent des différents constructeurs : chaque composant possède ainsi sa mesure. Par conséquence, obtenir une vue d'ensemble homogène d'un système hétérogène devient un exercice complexe, tant au niveau analytique que pratique. Pour placer l'utilisateur au sein du processus d'évaluation de la qualité de service, il faut donc en plus des indicateurs traditionnels, un nouveau modèle de QoS. Ce modèle comprend ainsi une nouvelle couche d'évaluation de la QoS soumise à la perception de la QoS par l'utilisateur final (voir Figure 1-1).

[3] ping - commande **DOS** pour tester l'adresse IP de destination que vous souhaitez atteindre et pour enregistrer les résultats

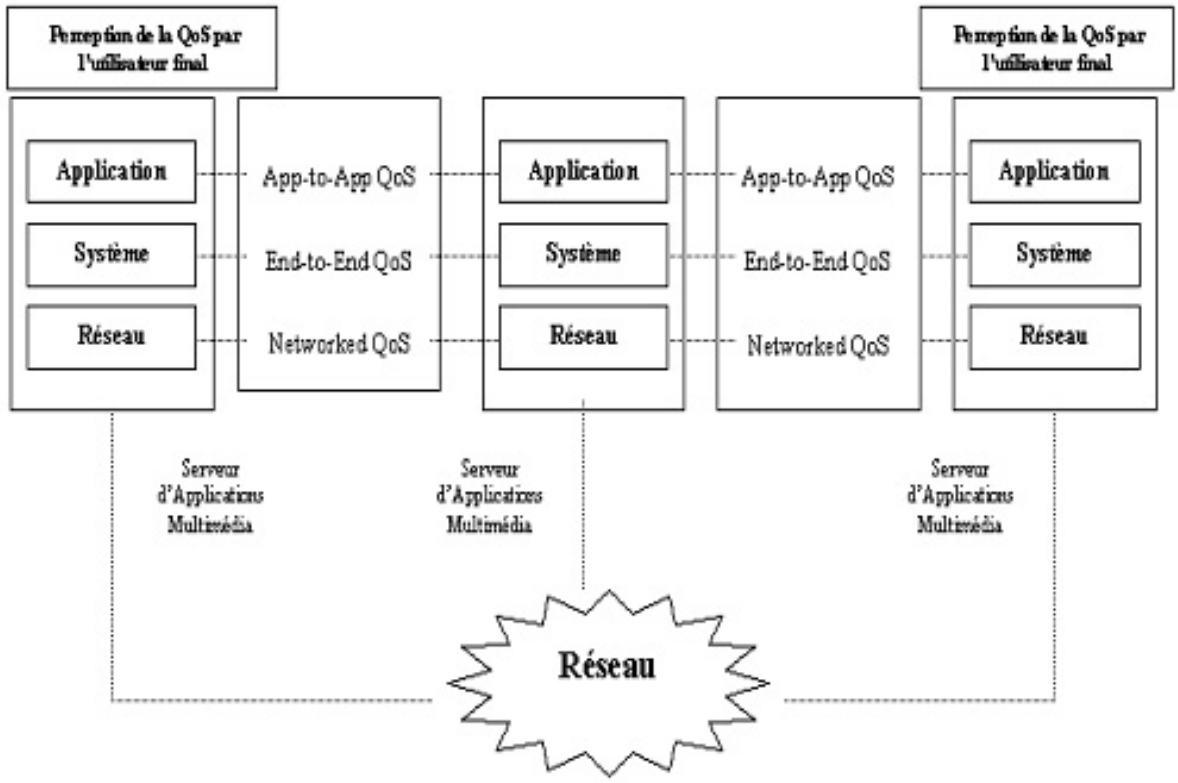


Figure 1-1 Environnement Distribué avec les différentes couches de la QoS

Pour compléter le modèle Nahrstedt en 3 couches (application, système et réseau) de la QoS [Réf. NAHRS], nous avons une quatrième couche, celle de l'utilisateur final. Donc les couches du nouveau modèle peut être décrit comme suit :

1. *Un niveau réseau* où nous trouvons des caractéristiques de qualité telles que la largeur de bande, le retard, le jitter et le taux de perte de paquets.

2. *Un niveau système* qui couvre le système d'exploitation et de communication ainsi que les dispositifs associés. Ici les principales caractéristiques de la qualité sont la disposition du CPU[4], son utilisation, les paramètres de stockage, les mécanismes de fonctionnement de la mémoire tampon.

3. *Un niveau application* avec comme paramètres de QoS par exemple, pour des images : la taille d'image, la cadence des trames, le délai et la fiabilité.

[4] CPU – Acronyme pour Central Processor Unit qui représente la partie de l'ordinateur qui contient des circuits pour l'interprétation et l'exécution des instructions

4. Un nouveau *niveau utilisateur* avec fondamentalement deux grandes familles de types de paramètres qui sont liés à l'usage :

- le premier type concerne des paramètres de la QoS dans le cas d'une communication avec un serveur (exemple le World Wide Web ou le streaming vidéo) ;
- le deuxième type concerne des paramètres de la QoS dans le cas d'une communication entre deux ou plusieurs utilisateurs (le courrier électronique ou la vidéoconférence).

L'ensemble des paramètres de la QoS à ce niveau représentent des paramètres QoS des couches inférieures d'un réseau perceptibles à un point d'accès de service et des paramètres additionnels qui sont liés à l'utilisation du service.

Selon [Messerschmitt] (tableau 1.1), nous pouvons établir une taxonomie spécifique des applications réseau qui correspond au *niveau utilisateur*. Ces applications, typiques des environnements distribués, combinent des communications client-client ou client-serveur en mode immédiat (synchrone, temps réel) ou différé (asynchrone).

	Immédiat	**Différé**
Client – Client	Téléphonie Vidéoconférence	E-mail Voice mail
Client – Serveur	Vidéo sur demande WWW sur le Net	Transfert de fichiers

Tableau 1-1 Une taxonomie des applications réseau.

Dans cette thèse, nous nous concentrons sur ce nouveau niveau (niveau utilisateur final) et nous nous intéressons principalement à la perception de la qualité du service par l'utilisateur final sans oublier les autres couches de QoS. Nous fournirons un cadre nécessaire pour que les utilisateurs puissent faire l'évaluation de la qualité de service. Ce que voit l'utilisateur est le sommet de l'iceberg de la mise en place du service. Heureusement, il est possible de limiter la complexité et le nombre des caractéristiques de la qualité à un petit nombre de paramètres, ce qui aide à automatiser le processus de l'évaluation de la qualité de service. Cette thèse présente une approche pour

résoudre les problèmes de l'évaluation de la qualité de service vu par l'utilisateur dans les environnements distribués.

1.2 Objectifs de cette thèse

Le but de cette thèse est d'apporter des outils pour aider les utilisateurs à effectuer l'évaluation de la qualité de service. L'approche choisie est la définition d'un nouveau cadre pour l'évaluation de la qualité de service qui porte sur :

- Une procédure d'évaluation de la qualité de service en 5 étapes avec la participation de l'utilisateur final ;
- Une architecture pour l'évaluation de la qualité de service contenant des méthodes d'évaluation appliquées par des agents intelligents, et finalement des applications et des services à l'essai ;
- Une plate-forme d'expérimentation de l'architecture d'évaluation de la qualité de service pour donner aux utilisateurs l'orientation nécessaire et un environnement de confiance pour leurs tests d'évaluation.

Par rapport au modèle que nous avons établi dans le paragraphe précèdent (voir Figure 1-1), cette thèse se concentre sur les aspects de la couche application et service. En particulier, elle cherche à répondre aux questions suivantes :

- Quelle évaluation de la qualité de service pour des applications réparties du point de vue d'un utilisateur ?
- Quelles sont les déficiences et les manques des méthodes de test de la qualité de service et des produits de gestion de niveau de service du point de vue de l'utilisateur final ?
- Comment ces déficiences peuvent-elles être compensées par une nouvelle méthodologie et une procédure commune pour l'évaluation de la qualité de service par rapport à différents services ?

Ainsi, cette thèse aborde les questions suivantes:

- La qualité perçue par l'utilisateur du service n'est plus seulement une question de réseau, mais également une question d'application (voir la section 2.2).

- Les méthodes de test de la QoS et les produits de gestion de niveau de service sur le marché aujourd'hui ne sont pas suffisants (voir la section 2.4").

- L'évaluation de la qualité de service lancée par l'utilisateur peut être automatisée (voir la section 3.1).

- La participation des utilisateurs à l'évaluation de la qualité de service est possible (voir la section 3.2).

- Le procédé d'évaluation de la qualité de service proposé peut être appliqué aux applications et aux services divers à l'aide des Agents Intelligents (voir la section 5.1).

- Ouverture vers les opérateurs de télécommunication, les constructeurs ou les fournisseurs de service (voir la section 6.4).

La méthodologie proposée sera décrite en détail et appliquée à un serveur mail, aux services en ligne (WWW), et à une application en temps réel, le streaming vidéo.Un exemple d'exécution prouvera sa viabilité.

Enfin cette approche devrait également être intéressante pour les fournisseurs de services qui vont se présenter sur le marché notamment dans le domaine des mobiles et des applications multimédia[1].

1.3 Importance de cette recherche

Cette thèse est principalement destinée aux utilisateurs essayant d'obtenir des résultats sur l'évaluation de la qualité de service en appliquant des méthodes simples et générales tout en tenant compte des caractéristiques essentielles de la qualité. Cette thèse leur offre une méthode, un outil et un environnement pour réaliser cette évaluation.

Ce cadre peut être un point de départ pour une standardisation, pour définir un ensemble d'acceptions et d'authentifications des outils (c.-à-d., méthodes d'évaluation), pour l'évaluation de la qualité de service en tant qu'élément de l'infrastructure de la mise en place d'un service.

[1] Afin de trouver une évaluation commune au sujet de qualité de service, il est raisonnable que les fournisseurs de service et les utilisateurs de services appliquent les mêmes méthodes d'évaluation de la qualité des services.

Avec l'évolution de leurs réseaux les opérateurs sont obligés de s'intéresser de plus en plus à la qualité de service. Le cadre de cette thèse peut être un point de départ pour une démarche «qualité de service – clients ».

La participation des utilisateurs est directement liée à l'existence d'un ensemble harmonisé d'outils pour produire une évaluation de la QoS. Cette évaluation doit être conséquente et cohérente avec les coûts de l'infrastructure relative à la mise en place du service. Si nous ne pouvons pas la mesurer, la qualité n'existe pas ! Cette compréhension de base peut faire adopter l'évolution des mesures de la qualité de service.

1.4 Contribution personnelle

La principale contribution de cette thèse est de proposer une approche d'évaluation de la qualité de service vue par l'utilisateur final. Les principales apports sont :

- ***proposition d'un cadre général d'évaluation de la qualité de service***

Un cadre complet qui contient un modèle d'un mécanisme ainsi qu'une procédure d'évaluation de la qualité de service a été conçu dans cette thèse. Le modèle comporte un Utilisateur, un Registre qui est une base de données propriété d'un opérateur ou d'un fournisseur de service mobile, une Implémentation et naturellement une application/service pour le test. Ce modèle est centré sur l'utilisateur final qui détient le rôle principal dans notre démarche. C'est lui qui décide quand il peut faire des tests d'évaluation de la qualité de service. Il participe d'une manière indirecte par ses résultats, ses expériences et ses opinions à l'amélioration de la « boite à outils d'évaluation QoS ».

La procédure d'évaluation QoS que nous avons proposée en 5 étapes est conforme à la norme ISO 9000 et s'adresse à des utilisateurs non techniques.

La première étape consiste à identifier tous les paramètres essentiels et non essentiels de la qualité [***identification***] à partir de différents documents spécifiques. La deuxième nous permet de distinguer les paramètres essentiels [***classification***] des paramètres non essentiels. Ensuite parmi les paramètres essentiels, nous devons choisir ceux que l'on peut mesurer [***sélection***] et quantifier [***quantification***] afin de donner des valeurs de seuils correspondantes. Les valeurs obtenues de différentes mesures seront ensuite comparées à ces seuils [***évaluation***].

- ***une architecture d'évaluation automatique de la QoS***

Au cours de cette thèse, nous avons argumenté le fait que l'évaluation de la qualité de service doit être automatisée pour obtenir des résultats utilisables. Nous avons donc proposé une structure avec des Agents Intelligents pour faciliter et automatiser cette évaluation. Cette structure est composée d'un *Agent Manager* qui est directement lié au *Registre* (base de données de l'opérateur) et qui dirige les différents *Agents* (Task Agent) et *l'Interface* avec l'utilisateur (l'interface elle-même est représentée par un Agent). L'environnement de travail de ces Agents est une plateforme Jade LEAP dans notre cas, leur moyen de communication étant le langage ACL (Agent Communication Language) adapté à leurs besoins et leurs ontologies. La très forte adaptabilité des Agents Intelligents leur permet d'être en adéquation constante avec les évolutions des services proposés par les opérateurs où les fournisseurs de services.

- ***application de la procédure d'évaluation de la qualité de service***

Nous avons testé notre approche sur 3 services : l'accès web du type www, le mail et le streaming vidéo pour des fichiers MPEG 1. Ces services ont été choisis suite à des études de marché réalisées avec les étudiants de l'INT. Nous avons appliqué la procédure en 5 étapes pour les 3 services. Nous avons obtenu ainsi les paramètres essentiels et mesurables avec des valeurs de seuils précises.

- ***un prototype d'implémentation de l'évaluation de la qualité de service avec des Agents Intelligents***

Nous avons évalué dans un premier temps les plates-formes qui utilisent les Agents Intelligents. Notre choix c'est porté sur la plate-forme Jade-LEAP qui propose un ensemble assez complet d'Agents et des exemples sur lesquels nous nous sommes appuyés dans notre développement. Jade LEAP est composé de 2 parties : la plate-forme Jade et des bibliothèques LEAP. Les Agents communiquent par le langage FIPA ACL. Un éditeur est disponible pour l'enregistrement et la gestion des Agents. Aucune interface n'est disponible pour le développement et l'implémentation. Pour combler cette lacune, nous avons utilisé le J2ME Wireless ToolKit qui propose des interfaces proches des téléphones mobiles actuels. La facilité de programmation et la présence d'une machine virtuelle intégrée dans le langage Java permettent une aisance au niveau du développement des Agents. De plus l'environnement dans lequel nous avons travaillé, est un environnement distribué qui correspond à l'environnement des réseaux d'accès mobiles.

Le code est ouvert et extensible vers d'autres services comme le VOD (Video On Demande) ou le ftp [Réf 959] qui se développent de plus en plus.

1.5 Organisation du document

Le mémoire de thèse est organisé de la façon suivante :

Le chapitre 2 commence avec la terminologie requise pour comprendre la qualité de service dans les environnements distribués. Nous présentons ensuite l'état actuel des réseaux mobiles et les contraintes en terme de QoS. Cette présentation permet de mieux comprendre et de trouver les conditions d'un cadre d'évaluation de la qualité de service du point de vue des utilisateurs. Ces conditions sont comparées aux solutions proposées dans les normes de standardisation internationales, dans les travaux scientifiques, et dans les produits du marché. Les lacunes sont décrites ensuite.

Le chapitre 3 présente un nouveau modèle d'évaluation de la QoS qui se focalise sur l'utilisateur final. Il se compose d'une nouvelle procédure d'évaluation, de son architecture et de son support d'exécution. Le même chapitre introduit le domaine des Agents Intelligents et les raisons qui nous ont conduit à les utiliser. Afin de prouver que différentes caractéristiques, normes, et recommandations peuvent être employées avec notre procédure d'évaluation, nous employons un langage connu de programmation (Java) et également un ensemble de caractéristiques techniques en tant que point de départ.

Dans le chapitre 4, nous décrivons l'application de la procédure dévaluation de la qualité de service présentée dans le chapitre 3, pour 3 services multimédia repartis: le service de courrier électronique, les services web en ligne (www) et les services de streaming vidéo MPEG-1. Afin de prouver que notre processus d'évaluation est fiable, les caractéristiques, les normes et les recommandations différentes peuvent être employées en tant que point de départ. Les exemples numériques donnés, basés sur la seule expérience de l'auteur, servent seulement d'illustration. C'est l'utilisateur qui, finalement, doit définir selon différentes conditions et contraintes les paramètres de la qualité qu'il souhaite tester.

Le chapitre 5 décrit une exécution de l'évaluation de la qualité de service en utilisant la technologie d'agent. Nous présentons un Toolkit Multiagents appelé ***Jade-LEAP*** et un prototype d'agent d'évaluation de la QoS pour des services en ligne (www), le courrier électronique et le streaming vidéo. Ce chapitre vise à prouver que des agents peuvent servir à une évaluation automatisée de la qualité de service pour différentes applications et services avec une interface proche du téléphone

mobile actuel. Ils réalisent les 5 étapes de notre procédure d'évaluation de QoS. L'utilisation des agents mène à des résultats comparables, objectifs, et reproductibles et surtout ils sont faciles à manipuler par les utilisateurs des services. Ce chapitre est une preuve du concept et il fournit des informations utilisables pour une réelle implémentation d'évaluation de la qualité de service en tant qu'outil de test dans un réseau de téléphones mobiles.

Le chapitre 6 apporte des conclusions issues de l'approche théorique, la réalisation et du test de prototype. Nous présentons également l'impact potentiel du prototype au niveau des opérateurs, constructeurs et fournisseurs de service. Les questions ouvertes concernant la QoS, les problèmes qui n'ont pas trouvé de solutions ainsi que les perspectives du travail effectué sont identifiés.

2 Nécessité d'une nouvelle évaluation de la QoS

Le chapitre 2 *s'ouvre avec la terminologie requise pour comprendre la notion de qualité de service dans les environnements distribués. Suit une courte présentation des réseaux actuels mobiles et leurs contraintes en terme de QoS. Celle ci nous permettra d'établir les conditions de la mise en place d'un cadre d'évaluation de la qualité de service du point de vue des utilisateurs. Ces conditions sont ensuite comparées aux solutions proposées dans les normes de standardisation internationales, dans les travaux scientifiques, et celles du marché. Leurs lacunes sont ensuite décrites et expliquées. Nous identifions ainsi le besoin d'un nouveau cadre d'évaluation de la qualité de service.*

2.1 Terminologie

Cette section définit les principaux termes et expressions liés à la notion de QoS dans le domaine des applications.

- **La Qualité**

Il y a plusieurs définitions de la qualité. Nous pouvons citer :

Ensemble des caractéristiques d'une entité qui lui confèrent l'aptitude à satisfaire des besoins exprimés et implicites [Réf. ISO 8402].

Obtention de la satisfaction durable du client, en répondant à ses besoins et attentes, au sein d'un organisme s'engageant à améliorer constamment son rendement et son efficacité (conceptions les plus récentes de l'ISO9000)

La notion de qualité, est une expression de la conformité pour l'utilisation. La conformité pour l'utilisation n'est pas déterminée par le producteur ou le vendeur, mais par le consommateur [Réf. ISO 8402]).

La qualité peut être soit une forme physique pour l'usage, soit une forme physique pour l'objet, elle peut aussi s'appliquer à la satisfaction du client ou qualifier la conformité à des exigences [Réf. E.800].

Dans cette thèse, nous nous concentrons sur la dernière interprétation. Nous considérons que l'entité est un service multimédia ou une application répartie.

- **La qualité de service du service (QoS)**

Un service est le résultat produit par des activités à l'interface entre le fournisseur et le client et par des activités internes des fournisseurs pour satisfaire les besoins des clients [Réf. IS08402]. C'est l'ensemble des fonctions fournies et accessibles aux clients.

[E. 800] définit la qualité de service *comme étant l'effet global procuré par la qualité de fonctionnement d'un service qui détermine le degré de satisfaction de l'utilisateur*. Les propriétés de non fonctionnalité d'un service sont désignées aussi sous le nom de propriétés de la Qualité de service.

- **La qualité de service d'un réseau**

La qualité de service d'un réseau servait à l'origine à désigner une différenciation dans la qualité d'acheminement des données circulant sur les réseaux Internet. En termes plus simples, la qualité de service devait tendre à ce que le trafic dit "prioritaire" ait plus de chances d'atteindre sa destination que le trafic "normal". Aujourd'hui, ce concept couvre un champ beaucoup plus précis : la qualité de service désigne avant tout un transfert à débit garanti entre l'émetteur et le récepteur des données, et ce avec des temps d'attente (de latence) réduits au minimum.

Au moins deux raisons confirment l'importance de ce concept. La première est que dans les périodes de congestion du réseau, il est primordial de trouver un mécanisme qui autorise un traitement différencié des données en circulation (nous pouvons parler de classes de service). La seconde est que par ce biais, les fournisseurs d'accès trouvent le moyen de proposer des services à valeur ajoutée à leurs clients afin de se distinguer de leurs concurrents.

- **Les caractéristiques de la QoS**

Afin de fournir les spécifications complètes d'un service multimédia, il est nécessaire de considérer également les propriétés de non fonctionnalité associées à un service. Ces propriétés, désignées sous le nom de caractéristiques de la qualité de service, constituent l'aspect quantifiable de la QoS, indépendant du moyen par lequel il est représenté ou commandé.

- **La gestion de la QoS**

La gestion de la QoS représente n'importe quelle activité exécutée par un système ou un service de communications pour effectuer la surveillance, la commande et l'administration de QoS [Réf. X.641].

- **L'évaluation de la QoS**

L'évaluation de la qualité est l'examen systématique en vue de déterminer dans quelle mesure une entité (par exemple, un processus, un produit, un service) est capable de satisfaire aux exigences spécifiées [IS08402].

- **La détermination de la QoS**

En comparant la mise en place d'un service aux besoins de l'utilisateur, on obtient la détermination de la QoS, qui exprime le succès ou l'échec du respect des garanties convenues de la QoS.

- **La mesure de la QoS**

La mesure de la QoS est l'observation de la QoS ; c'est la méthode appliquée comme préalable à l'évaluation de la qualité. La mesure de la qualité permet de mesurer et de visualiser les niveaux de la qualité atteinte. Un système qui mesure la métrique de la qualité peut être considéré comme *une échelle quantitative et une méthode utilisables pour déterminer la valeur de la qualité d'un dispositif pour un produit logiciel spécifique [Réf. IS0 9126]*. La métrique mesure les attributs quantifiables spécifiques de la QoS dans les applications, les systèmes et les ressources.

Conformément à [Réf Fouq] une méthode pour mettre en place la mesure de la qualité de service consiste à « remonter » des variables (caractéristiques de la QoS) aléatoirement dans le temps. En pratique, les plates-formes d'administration sont utilisées dans ce contexte. Avantage notable : tous les centres d'exploitation réseau disposent d'une telle plate-forme. Ce type de mesures occasionnelles peut également s'avérer utile pour prendre connaissance de dysfonctionnements ponctuels du système. En revanche cette pratique ne permettra jamais d'obtenir des informations sur plusieurs mois ou plusieurs années et d'offrir une bonne perception, par service d'un groupe de ressources.

En ce qui concerne les services grand public, le volume des mesures est trop important pour être traité à la volée dans son ensemble, dans des délais acceptables. La mesure a alors lieu sous forme d'échantillonnage, approche indispensable pour observer des données de grand volume. L'analyse des flux des appels des mobiles vers les mobiles, avec indication géographique génère une telle

quantité d'informations sous forme de fichier log (plusieurs gigaoctets par heure), qu'il est pratiquement impossible de « remonter », traiter et analyser toutes ces données de façon suffisamment rapide pour les utiliser.

- **Service Level Agreements (SLAs)**

Les SLAs représentent des accords de garantie de la QoS. Aux termes de ces accords, l'opérateur s'engage contractuellement sur une qualité de prestations sous peine de pénalités financières. Certains SLAs sont attachés aux contrats types des opérateurs, d'autres sont négociés sur mesure notamment avec les grands comptes.

La rédaction d'un SLA avec un opérateur doit être aussi précise et explicite que possible. Il s'agit d'éviter tout malentendu, mais aussi de définir des critères objectifs facilement mesurables, qui serviront de base au suivi de l'exécution du SLA. En général, les opérateurs fournissent des rapports et des statistiques pour rendre compte de leurs performances. Les rapports sont souvent dépourvus d'intérêt, et il est difficile de comparer les résultats fournis avec les critères retenus pour un SLA. Pour superviser un SLA complexe, il est généralement nécessaire d'investir dans de coûteux équipements et logiciels, sans oublier les dépenses en personnel.

Actuellement les pénalités dues par les opérateurs justifient à elles seules le SLA. Or, les pénalités standards sont ridicules par rapport aux enjeux. Un engagement non tenu donne lieu bien souvent pour les grands comptes à un remboursement de deux ou trois jours du prix du contrat mensuel, ce qui est sans commune mesure avec le préjudice subi lorsqu'une panne a empêché le fonctionnement d'un élément vital !

Aujourd'hui, pour un utilisateur lambda, même si sur les réseaux fixes quelques soucis sont apparus récemment, les critiques concernent surtout les réseaux mobiles. Les problèmes de transmission des SMS entre les opérateurs, les dégradations progressives des communications dans le temps, les manques de capacité des communications fixes vers les mobiles sont des problèmes concrets auxquels les opérateurs sont incapables de répondre. Aucune garantie n'est envisageable dans un futur proche. Cependant, avec l'approche de l'UMTS et l'augmentation de la complexité des services, il est probable que les opérateurs signeront des SLAs avec n'importe quel utilisateur aussi bien sur des services basiques comme la voix ou la messagerie que sur des services plus complexes comme le téléchargement de gros fichiers ou la vidéoconférence.

- **Service Level Management (SLM)**

La gestion du niveau de service (SLM) est l'ensemble des procédures et des outils qui permettent de mesurer et contrôler la qualité permet en général du réseau et du Système d'Information (SI). Pour en mesurer le niveau de qualité, l'entreprise ou un prestataire dispose généralement des sondes qui viennent surveiller, mesurer, détecter, diagnostiquer les problèmes, alerter par e-mail et/ou émettre des rapports d'activité mensuels. Son fonctionnement vise à la fois les clients extérieurs et les utilisateurs internes.
« Le SLM englobe le système et la gestion du réseau et représente donc un cadre plus large pour développer et exécuter les accords de niveau de service (SLAs) » [Réf. Engel].

- **Class of Services (CoS)**

Les classes de service peuvent être comparées avec les classes proposées par les compagnies aériennes. Les services offerts et les tarifs sont complètement différents d'une classe à une autre. Les classes de service sont définies de façon à offrir des garanties minimales de QoS à chacun des types d'application qui seront transportés par le réseau.

2.2 Les exigences des utilisateurs

L'évaluation de la Qualité de service par l'utilisateur final constitue le sujet principal de cette thèse. Avant toute proposition de cadre, procédé ou outil de simulation, il convient de définir et de préciser les exigences des utilisateurs. Les mécanismes ou les méthodes d'évaluation de la qualité de service d'un réseau ne garantissent pas la qualité de service perçue par un utilisateur. La situation est en fait plus complexe (Figure 2-1). Indépendamment de la qualité du réseau, d'autres facteurs tels que la plate-forme de service, le logiciel utilisé et l'infrastructure locale ont une influence sur la qualité perçue par l'utilisateur. Les différentes parties (individus, organismes, fournisseurs) sont impliquées et responsables. Prenons l'exemple simple d'un service, l'accès web avec un terminal mobile. En fait la qualité service perçue par l'utilisateur final dépend de la qualité du réseau (réseau opérateur mobile), de la qualité de la plate-forme du service (cette plate-forme appartient au fournisseur de service), de la qualité du logiciel d'accès au service (http, ftp etc.) et bien évidemment de la qualité du terminal (infrastructure locale). En fait, l'utilisateur n'est pas du tout intéressé par les détails techniques de la disponibilité du service, seule sa perception individuelle est importante.

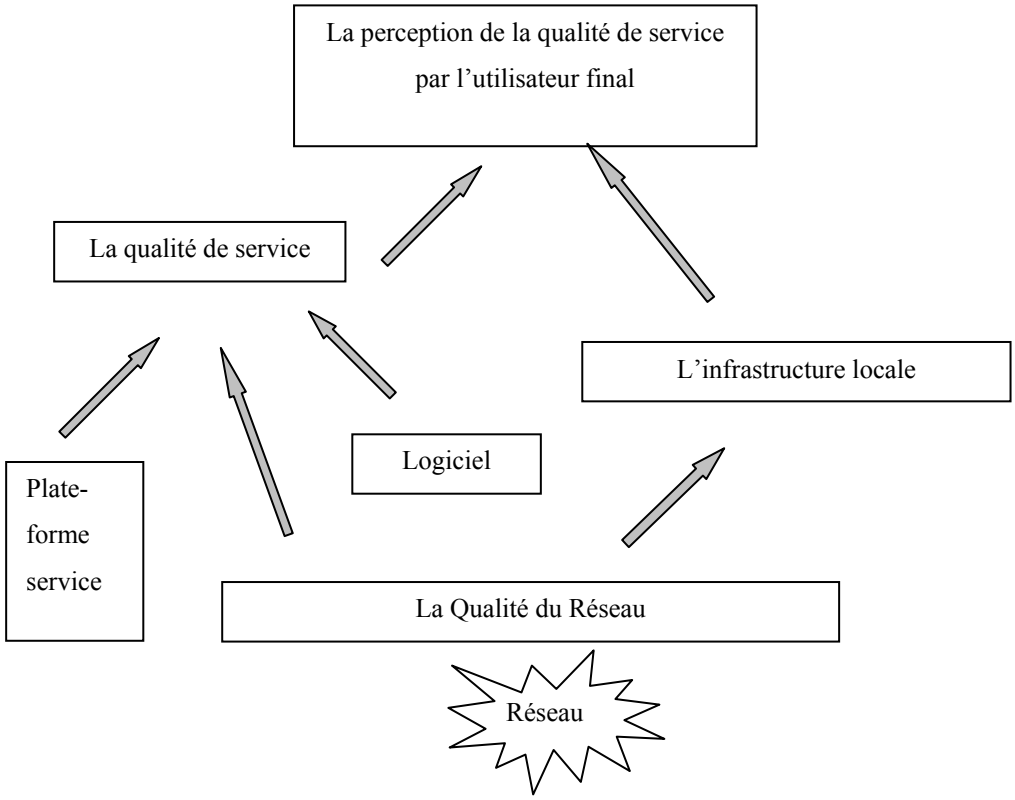


Figure 2-1 Les Dépendances de la Qualité [Bissel00a].

Pour les fournisseurs de service, l'amélioration de la qualité exige une évaluation continue de tous les processus de service. L'information nécessaire pour ce processus d'amélioration est fournie par des vérifications du fonctionnement du réseau interne et par l'évaluation des clients. Les fournisseurs de service doivent communiquer avec leurs clients au moment de l'achat d'un service mais aussi pendant toute la durée de celui-ci, voire après coup. L'efficacité de la communication avec des clients implique [Réf. IS0 9004]:

- une description du service, de sa portée et de sa disponibilité ;
- une estimation du coût du service ;
- des explications à propos des corrélations entre service, livraison et coût ;
- des explications relatives aux difficultés et problèmes susceptibles de se produire, à la manière de les résoudre en cas de besoin ;
- une assurance que les clients se rendent compte de la contribution qu'ils peuvent apporter ;

- la fourniture d'équipements proportionnés et aisément accessibles pour la communication efficace ;

- une décision sur le rapport entre le service offert et le besoin réel du client.

2.2.1 Quels sont les différents types de besoins et d'exigences du client?

Nous partons du principe suivant :

« L'organisme détermine les besoins et attentes des clients ainsi que les exigences imposées, puis les convertit en exigence spécifiées » [Réf. ISO 9000 :2000].

Nous distinguons trois types de besoins et attentes du client :

- exprimés : ceux explicités clairement par le client aussi appelés exigences du clients,

- implicites : ceux qui ne sont pas formulés mais que l'organisme, conscient des règles de l'art, de sa connaissance du produit et de son utilisation et de ses responsabilités doit honorer ;

- latents (ou potentiels) : ceux dont ni le client ni l'organisme n'ont encore pris connaissance mais qui deviendront importants dans l'avenir (par exemple, des besoins et des attentes liés aux évolutions technologiques ou du marché).

Aux yeux des clients, certains besoins sont considérés comme indispensables. Ce sont des demandes auxquelles le fournisseur n'échappera pas, et qu'il faut absolument satisfaire. Ce sont des fonctions que le client considère comme allant de soi et qu'il n'imagine pas ne pas voir satisfaites.

Nous distinguons trois types d'exigences relatives au produit :

- du client : besoins et attentes clairement exprimés par le client,

- imposées : exigences réglementaires, légales, techniques, normatives, etc.,

- spécifiées : exigences résultant de la spécification des besoins et attentes des clients et des exigences imposées.

2.2.2 Pourquoi identifier les besoins et attentes des clients?

Tout client a des besoins et des attentes mais n'est pas toujours capable de les exprimer en totalité. L'organisme déchiffre et interprète ces besoins et attentes et les complète, le cas échéant. Pour répondre à la demande, l'organisme crée et réalise un produit nouveau, améliore ou fait évoluer un produit déjà existant ou choisit un produit existant qui répond aux exigences du client. L'organisme fournit ensuite ce produit à son client qui le compare à ses besoins et attentes initiales (Figure 2-2).

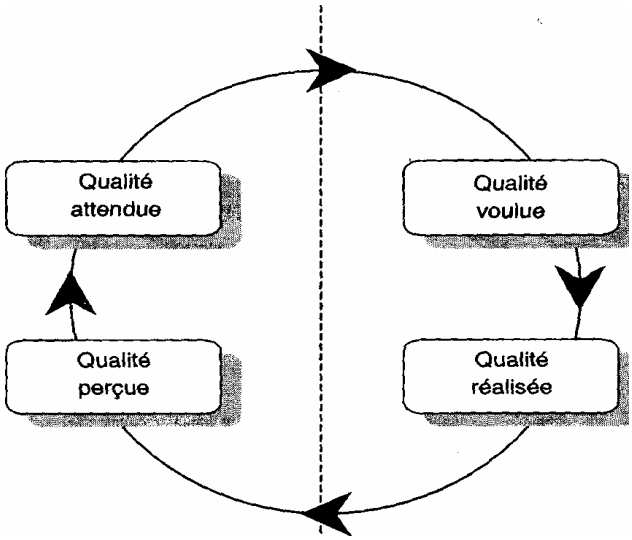


Figure 2-2 Différents types de Qualité [AFNOR]

Du point de vue de l'organisme, nous distinguons :

- la qualité **voulue** : caractéristique qualité que l'organisme souhaite atteindre pour répondre à la qualité attendue. C'est la prestation qu'il veut fournir à ses clients.

- la qualité **réalisée** : caractéristique qualité réellement réalisée par l'organisme.

Dans un monde parfait, la qualité réalisée serait identique à la qualité voulue. Ceci est difficile à obtenir en raison des dysfonctionnements et des aléas de toutes sortes qui créent des écarts entre ce qu'on prévoit ou planifie et ce qu'on réalise. On mesure donc l'écart par des indicateurs qualité mis en place dans l'organisme. L'écart minimum correspond à la conformité aux exigences imposées par des référentiels, comme la norme ISO 9001. La reconnaissance officielle de la qualité réalisée est

un facteur dynamisant et stimulant pour l'organisme, entraînant et sécurisant pour les différents clients.

Du point de vue du client, nous distinguons :

- la qualité **attendue** : caractéristique qualité souhaitée par les clients, c'est-à-dire la réponse à leurs besoins et attentes. Cela signifie que les conditions d'obtention de ce niveau de qualité ont été prédéfinies.

- la qualité **perçue** : caractéristique qualité ressentie, de façon plus ou moins confuse par le client à partir de ses propres besoins et attentes. Elle est l'expression de sa satisfaction, avec ce que cela comporte de subjectivité et d'impondérable. L'organisme évalue le degré de satisfaction de ses clients en mettant en place des outils de mesure de la satisfaction des clients et de la performance du service rendu.

L'écart est mesuré par des évaluations auprès des clients.

- un écart négatif reflète une insatisfaction des clients ;

- un écart nul reflète une satisfaction minimale des clients ;

- un écart positif permet à l'organisme de satisfaire ses clients, condition nécessaire pour les fidéliser.

Dans un monde parfait la qualité perçue serait identique à la qualité attendue. Ceci est irréalisable à cause de l'importance de l'irrationnel et du subjectif des clients dans la conscience de la qualité attendue et dans la perception de la qualité réalisée.

La satisfaction du client passe donc, entre autres, par l'étape fondamentale d'identification de ses besoins et attentes en termes de qualité attendue. Le but de cette opération est de réduire le différentiel entre la qualité attendue par le client et la qualité perçue lors de la fourniture du produit et d'optimiser le processus d'élaboration du produit.

Pour faciliter la participation des utilisateurs à l’évaluation de la Qualité de service et garantir les résultats de l'évaluation, le processus de l’évaluation de la Qualité de service doit être automatisé. Dans ce cas, seules quelques caractéristiques importantes doivent être sélectionnées et évaluées afin de limiter la complexité du processus. Se rendre compte de la problématique et comprendre précisément les informations techniques relatives au service fourni permettent à un utilisateur de

jouer un rôle et de se forger une opinion. Ceci nous amène automatiquement aux exigences et aux besoins différents.

Comme utilisateur des différents services et après avoir travaillé sur différents études de marché avec les étudiants de l'INT (voir dans le cadre d'un module de formation marketing dit EM 21), j'ai pu à travers les remontées du terrain identifier et compiler une liste des exigences les plus souvent exprimées par les utilisateurs désireux de faire l'évaluation de la Qualité de service. Cette liste est présentée dans la prochaine section.

2.2.3 Liste des souhaits d'un utilisateur pour l'évaluation de la Qualité de Service

Dans la section 1.1 "Problématique et motivations de la thèse " nous avons insisté sur le fait que la positon de l'utilisateur final est importante dans l'évolution de la qualité de service de bout en bout. Ils souhaitent avoir un procédé d'évaluation de la QoS qui :

- couvre réellement la qualité que les utilisateurs perçoivent [Réf. I.350] ;
- adresse les caractéristiques de la qualité qui peuvent être directement observées et mesurées au point auquel le service est consulté par l'utilisateur [Réf. E.800] ;
- se focalise sur des aspects de la qualité tels qu'ils sont précisés par les utilisateurs ai moment de la mesure;
- est capable d'étudier les exigences et le comportement d'utilisateurs ;
- se relie à un service multimédia ou à une application répartie et est indépendante de la couche de gestion de réseau [Réf. I.350] ;
- est facile à utiliser, peut être automatisé, et est autonome à partir d'un certain niveau ;
- utilise des protocoles simples et des interfaces simples telles que le SMTP (pour le mail) ou http (WWW) pour la communication avec l'utilisateur et des protocoles complexes pour la communication avec le service à tester ;
- produit des résultats reproductibles, compréhensibles, dignes de confiance, et comparables ;

- peut être normalisé comme préalable à une acceptation pour une utilisation générale dans l'avenir.

A la lecture d'une telle liste, un utilisateur pourrait se sentir désorienté. Alors où pourrait-t-il trouver de l'aide ? Est-il le seul à se trouver dans ce cas ? Sans doute finira-t-il par faire appel à tel organisme international de standardisation de la qualité ou tel autre, spécialisé dans les produits pour l'évaluation de la qualité de service (c.-à-d., gestion de niveau de service) sur le marché. Quelque soit l'orientation prise, l'utilisateur essuiera malheureusement un échec.

Afin de pouvoir retrouver l'utilisateur final et ses exigences par rapport à la QoS, nous allons nous intéresser à l'état actuel des réseaux d'accès mobiles et à leurs contraintes en terme de qualité de service. C'est l'objectif du prochain paragraphe.

2.3 Les réseaux d'accès mobiles

Le système de radiotéléphonie cellulaire, après une première génération constituée uniquement de réseaux analogiques, ont vu l'arrivée des technologies numériques au début des années 1990 en Europe (GSM), au Japon (PDC) et aux Etats Unis (PCS). L'évolution du réseau GSM (dit de 2ème génération « 2G ») vers l'UMTS (dit de 3ème génération « 3G ») passe par des étapes intermédiaires comme le GPRS, HSCSD ou EDGE (dites « 2.5G ») qui seront présentés dans les paragraphes suivants.

La troisième génération UMTS attendue pour l'année 2002 a pris un retard considérable par rapport au Japon. En Europe, le seul réseau « 2.5 G » qui a commencé à être déployé, est le GPRS. L'offre commerciale actuelle des opérateurs cible plus les professionnels et les aspects traités concernant la qualité de service sont assez limités.

2.3.1 Les systèmes « 2G »

2.3.1.1 Le réseau GSM

Dans le cas des réseaux cellulaires GSM, le service fourni le plus important est le service voix et la satisfaction de **l'utilisateur** se traduit par trois contraintes de fonctionnement :

- L'abonné doit pouvoir joindre n'importe qui, n'importe quand et n'importe où ;

• Après établissement de la communication, la conversation est audible et compréhensible par les deux interlocuteurs ;

• La ligne téléphonique n'est pas coupée en cours de communication.

Pour **l'opérateur**, ceci se traduit par trois contraintes techniques :

• Une couverture nationale ;

• Un dimensionnement correct des liens radio et réseaux (pour la disponibilité des ressources) ;

• La mise en place de mécanismes efficaces de gestion du lien radio (pour la minimisation du taux de coupure).

Au quotidien, il n'y a pas de gestion de la qualité de service grâce à la nature du mode de connexion qui est le mode circuit. Dans ce mode, une ressource est dédiée à un seul utilisateur durant toute la durée de sa communication (en situation de faible mobilité). Comme indicateurs de la qualité de service, nous avons retenus ceux qui ont été introduites en 1998 par l'ART (Association de Régularisation de Télécommunications en France). [Réf. ART 2000].

Indicateurs	**Définition**
taux de communications réussies et maintenues 2 minutes	Une communication est considérée comme réussie si l'appel lancé par l'un des enquêteurs aboutit **dès le premier essai** et si la communication est **maintenue 2 minutes sans coupure**. Le taux est calculé sur la base du nombre total de mesures. Le complément à 100% est donc constitué du taux d'échecs après une tentative, plus le taux de coupures.
taux de communications de qualité auditive parfaite	Une communication est considérée comme parfaite si la qualité auditive perçue par les deux interlocuteurs est **parfaite** (comparable à la qualité des communications sur le réseau fixe).
taux de communications de qualité auditive acceptable	Une communication est considérée comme acceptable si la qualité auditive perçue par les deux interlocuteurs est **perturbée légèrement sans toutefois gêner la conversation.**
taux de communications de qualité auditive correcte	II s'agit de la somme des deux indicateurs précédents. Une communication est considérée comme correcte si la qualité auditive perçue par les deux interlocuteurs est **soit parfaite, soit acceptable.**
taux de communications réussies, maintenues 2 minutes et de qualité auditive correcte	C'est **le produit du taux de communications réussies et du taux de communications de qualité auditive correcte**. Son complément à 100% est le taux de communications perçues comme anormales : échouées ou coupées ou de qualité auditive médiocre ou mauvaise.

Tableau 2-1 Définition des indicateurs de la qualité de service.

Les taux de communications correctes, parfaites ou acceptables sont calculés sur la base du nombre de communications réussies (établies dès le premier essai et maintenues pendant 2 minutes). Les autres indicateurs se rapportent à l'ensemble des mesures effectuées.

Dans le rapport du Gouvernement au Parlement publié en juillet 2001, on trouve les chiffres exacts du taux de couverture et les obligations des 3 opérateurs mobiles en France (Figure 2-3). Nous pouvons constater une amélioration de la couverture depuis sous l'effet conjugué de la forte augmentation de la pénétration des services de téléphonie mobile et de l'importance de la couverture comme argument de vente. Le même rapport précise aussi que « *encore 8,4 % du territoire ne sont couverts par aucun opérateur mobile, soit approximativement l'équivalent de huit départements français. L'état de la couverture est encore insuffisant eu égard au besoin universel qu'est devenu aujourd'hui le GSM.* »

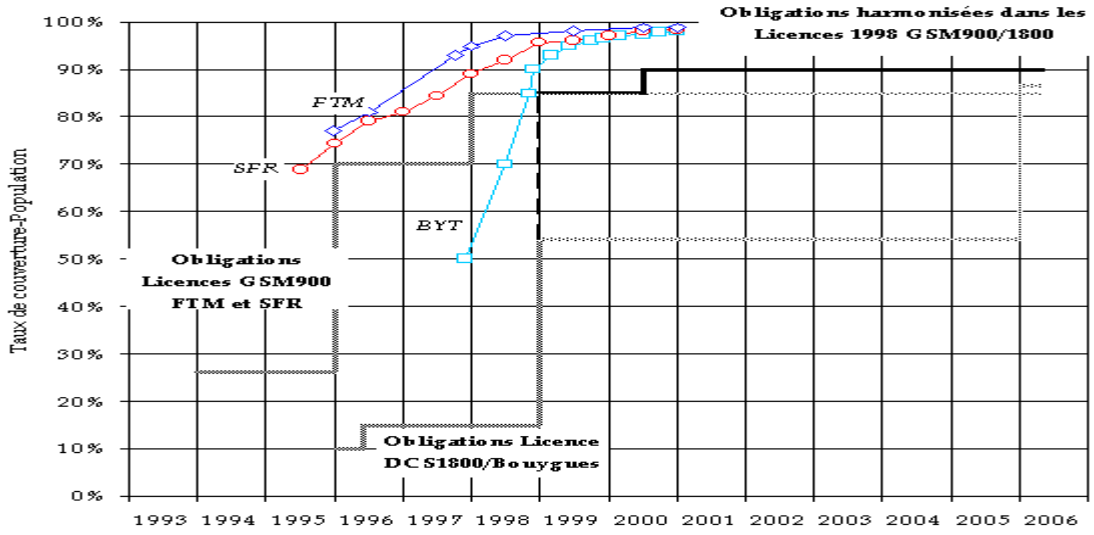


Figure 2-3 Taux de couverture réalisée et obligations des opérateurs mobiles.

2.3.1.2 Le WAP (Wireless Application Protocol)

Le *Wireless Application Protocol* ou Protocole pour applications sans fil, est un ensemble de standards qui permet de faire fonctionner des applications sur un terminal sans fil : les téléphones portables et les appareils de poche appelés assistant numériques personnels ou PDA utilisant les systèmes d'exploitation Palm OS, Epoc, Pocket PC (Windows CE), OS/9, Java OS, etc.

Point de convergence entre l'Internet et la téléphonie mobile, le Wap est adapté aux faibles débits des réseaux GSM actuels (9,6 kbps). Les protocoles Wap permettent de recevoir sur un téléphone portable des e-mails et de charger des pages web "allégées". Théoriquement, n'importe quel contenu Internet peut être adapté au Wap, seule la présentation change. Mais de plus en plus de fournisseurs de contenu et d'industriels proposent des services spécifiques, le plus souvent à destination des opérateurs télécoms.

En terme de QoS, nous nous retrouvons dans le même cas que pour le GSM. Les fournisseurs de service n'offrent aucune garantie sur l'accès et le contenu de leurs sites. De plus, leur faible contenu et les fautes en terme de marketing des opérateurs (seulement 30 000 abonnés en 2002) font craindre que le Wap ne se développera pas de manière spectaculaire.

2.3.2 Les systèmes « 2.5G »

2.3.2.1 Le GPRS (Global Packet Radio Service)

Le GPRS représente une évolution majeure du GSM. Par l'utilisation de communication par paquets et l'augmentation des débits, il ouvre la porte aux communications mobiles multimédia et permet la transition vers la troisième génération.

Un réseau GPRS peut être vu comme un réseau de données à part entière, qui dispose d'un accès radio réutilisant une partie du réseau GSM. (Réf GODLE/LAGRAN). Le réseau comprend des abonnés propres, mobiles ou fixes et peut être relié à différents réseaux de données par l'intermèdiare du protocole IP (Internet Protocol). Dans ce cas un terminal GPRS dispose d'un adresse IP dont le champ réseau est spécifique au réseau GPRS.

Les débits théoriques prévus (de 9.6 kbits/ à 171.2 kbits/s) permettent d'envisager des applications telles que la consultation du Web, le transfert de fichiers par FTP (File Transfert Protocol), la transmission de vidéo compressée, etc. En réalité le débit utile des terminaux sera de 10 kbits/s dans le sens mobile-station de base et de 30 à 40 kbits/s dans l'autre sens.

La qualité de service comprend plusieurs critères : la précédence, la fiabilité, le délai et le débit. Trois niveaux de précédence sont définis ; ils permettent à l'opérateur de gérer les situations de surcharge en interrompant le service des sessions moins prioritaires. Les niveaux de fiabilité correspondent à des garanties différentes sur des taux d'erreurs (cf. tableau 2.2) (GSM 02.60 table 41). Les différentes classes de délai sont indiquées dans le tableau 2.3.

Dans un premier temps, les réseaux offriront seulement un service de classe 4 *(best effort)* qui correspond à la classe assurée par le réseau Internet et les réseaux IP.

Classe	**Probabilité de Perte**	**Probabilité de duplication**	**Probabilité de déséquencement**	**Probabilité d'erreur**
1	10^{-9}			
2	10^{-4}	10^{-5}		10^{-6}
3	10^{-2}	10^{-5}		10^{-2}

Tableau 2-2 Classes de fiabilité de GPRS

Classe	**SDU[2] de 128 octets**		**SDU de 128 octets**	
	Délai moyen	Délai à 95 %	Délai moyen	Délai à 95 %
1	< 0,5 s	< 1,5 s	< 2 s	< 7 s
2	< 5 s	< 25 s	< 15 s	< 75 s
3	< 50 s	< 250 s	< 75 s	< 375 s
4	Non spécifié *(best effort)*			

Tableau 2-3 Classes de délai de GPRS

[2] SDU : Service Data Unit

2.3.2.2 HSCSD ou EDGE ?

EDGE (Enhanced Data rate for GSM Evolution) représente une seconde forme d'évolution des systèmes 2G. Le standard EDGE devrait constituer le point de convergence entre la famille GSM et AMPS aux Etats Unis. Il utilise une nouvelle modulation du signal (8PSK) qui permet d'améliorer l'efficacité spectrale et la capacité du réseau. EDGE est une évolution du système GSM et du système TDMA, mais son rendement optimal est obtenu lorsqu'il est combiné avec un réseau paquet (GPRS). Le débit peut aller jusqu'à 384 kbits/s.

Si la limitation de débit sur GSM est due au fait que le mobile transmet et reçoit sur un seul intervalle de temps par trame TDMA, il est envisageable de disposer de mobiles capables de transmettre et de recevoir sur plusieurs intervalles de temps (IT). Le service HSCSD (High Speed Circuit Switched Data) définit des circuits allant jusqu'à 6 IT allouées aux mobiles par trame TDMA. Les débits envisagés sont de 19.2 kbits/s, 28.8 kbits/s, 38.4 kbits/s, 48 kbits/s, 56 kbits/s ou 64 kbits/s suivant le nombre des canaux alloués avec cette technique.

Dans ces réseaux, la problématique de qualité de service est beaucoup plus complexe car nous sommes confrontés d'une part à un environnement multi-services et multimédia avec différentes exigences de bon fonctionnement et d'autre part à deux modes de transport de l'information : circuit ou paquet. De plus ces réseaux n'existent qu'en phase de prototype et donc les travaux de normalisation sont loin d'être finis. Il sera quasiment impossible de travailler sur la qualité de service sans avoir un réseau complet avec de « vrais utilisateurs ».

2.3.3 Le réseau UMTS (Universal Mobile Telecommunication System)

L'UMTS (Universal Mobile Telecommunications System) désigne une technologie de la famille dite IMT 2000 comme norme pour les systèmes de télécommunications mobiles dits de troisième génération.

Les technologies développées autour de la norme UMTS conduiront à une amélioration significative des vitesses de transmission avec des débits supérieurs à 144Kb/s et pouvant aller jusqu'à 2.Mb/s (en zone urbaine, avec une mobilité réduite). De tels débits sont significativement supérieurs à ceux permis tant par les réseaux GSM actuels (9,6Kb/s) que par leurs évolutions prochaines (GPRS à 30 Kb/s). Cette amélioration des débits est rendue possible par l'évolution des technologies radio qui autorisent une meilleure efficacité spectrale et l'exploitation de bandes de spectre de fréquences supérieures à celles utilisées par la technologie GSM.

L'UMTS permet en effet la convergence d'Internet et de la téléphonie mobile et ouvre le marché des services de "données mobiles", en mettant sur l'ensemble des territoires couverts "Internet au bout des doigts" de tout utilisateur disposant d'un appareil électronique communicant. [Gouv].

Les réseaux UMTS devraient ainsi capter une part significative de l'accès à Internet au cours de la prochaine décennie.

La combinaison de l'interactivité et de la mobilité permet d'imaginer l'essor de multiples services avancés. Nous pouvons distinguer trois grands types de services avancés en fonction de la clientèle: les professionels, les particuliers et les services universels. Ils constituent une base de différenciation des systèmes. Les spécialistes considèrent à partir de là, 6 niveaux. [Réf .Radu, Salgues].

- *Les services destinés aux professionels*

Les services avancés sont destinés à la clientèle d'affaire et sont principalement la création de réseaux privés virtuels et les services basés sur la localisation. Ainsi, certaines personnes de l'entreprise pourront avoir accès au réseau public et d'autres constituer un groupe fermé. Les services basés sur la localisation ont un caractère plus commercial. Par exemple, en appelant le numéro national d'une chaîne hôtelière, la communication aboutit à l'hôtel le plus proche. L'accès restreint (accès à des services payants) est un service souvent demandé par la clientèle d'affaire.

- *Les services destinés aux particuliers.*

Les demandes concernent principalement la tarification de proximité (facilité d'appel dans la zone de résidence), les restrictions d'abonnement, le service de ligne alternative pour séparer sur un même mobile les appels professionnels des appels privés (avec souvent des tarifications correspondantes).

- *Les services universels.*

L'acheminement forcé vers un opérateur permet à l'exploitant de maintenir ses revenus et d'avoir une relation plus rapprochée avec ses clients. Les services liés à l'opérateur sont des services qui sont en lien direct avec le service clientèle. Ceux-ci représentent des demandes de renseignements simples, la même demande avec ré-acheminement sur le numéro recherché, l'assistance aux clients en activant et désactivant des services.

Le service de base qui consiste à transmettre de la voix, on distingue six niveaux de services permettent d'obtenir des revenus :

- niveau 0 : la messagerie vocale et la facturation détaillée ;
- niveau 1 : le roaming élargi, la messagerie généralisée ;
- niveau 2 : l'accès à des services d'information et à l'Internet ;
- niveau 3 : les services interactifs (réservation de chambres d'hôtel ou de billets de transport, etc),
- niveau 4 : la numérotation abrégée, les cartes prépayées, les restrictions d'accès ;
- niveau 5 : l'indépendance des réseaux (liens avec la banque, etc.).

Dans le cas d'une nouvelle norme, la compatibilité ascendante des services devient un problème. L'UMTS sera donc introduit comme une extension du GSM plutôt que comme un nouveau système destiné à le remplacer, du fait de l'importance des investissements réalisés. Cette caractéristique poussera les nouveaux réseaux UMTS à offrir la gamme entière des services GSM. Ceci sera rendu encore plus complexe par le développement des normes intermédiaires comme GPRS. Ceci permettra pour les services GSM, y compris les nouveaux services de type GPRS, une concurrence réelle entre les nouveaux réseaux UMTS et les différents niveaux de réseaux GSM préexistants.

Dans la conception de la norme UMTS, on a distingué deux niveaux de services : les services traditionnels qui existent de fait dans les réseaux actuels et les services émergents qui sont un moyen de différencier UMTS de ces réseaux.

- *Les services traditionnels*

De nombreux services orientés données sont ou seront progressivement supportés par le GSM. En particulier les évolutions du GSM permettront une première étape vers la transmission haut débit et vers d'autres services (courrier électronique, télépaiement, transfert de fichiers, accès Internet). Toutefois, l'UMTS (384 kbit/s pour tout le monde en mode mobile et 2 Mbit/s en situation «fixe») fournira un meilleur compromis capacité/coût.

- *Les services émergents*

A moyen et long terme, l'UMTS s'adressera au grand public. Dans ce cadre, la grande majorité des services qui seront proposés alors est encore inconnue. Ces nouveaux services répondront certainement à trois exigences : le contenu multimédia (exemples : jeux, loisirs), la mobilité, la création de valeur ajoutée (il faut que le grand public soit prêt à payer le surcoût de ces services).

Dans le tableau 2.4 nous avons présenté un récapitulatif de l'évolution des services GSM vers l'UMTS.

Type de services	**Actuel GSM**	**Intermédiaire**	**UMTS**
Téléphonie	Voix	Voix de haute définition	Voix associée à l'image
Communication	SMS	Email, Fax, services de reconnaissance vocale	Passerelles entre les médias Agents électroniques
Information	Actualité, météo, finance	LAN, Internet, Intranet	Journaux électroniques Magazines électroniques
Affaires et transaction		Applications intégrées Synchronisation des calendriers	Monnaie virtuelle Catalogue
Loisirs			Film à la demande Jeux interactifs

Tableau 2-4 Evolution de services GSM vers l'UMTS.

Certaines de ces applications ont des contraintes de transmission temps réel et d'autres non ; certaines sont tolérantes aux erreurs de transmission et d'autres non ; certaines demandent des débits élevés....

De façon plus précise, chacune de ces applications se caractérise par trois attributs : le délai et la variation de ce délai, le débit et le taux de pertes d'information acceptable. Ces attributs sont les caractéristiques de qualité de service d'une application.

Le mode de transport de l'information est le facteur qui impacte ces paramètres. Il peut être de deux natures : circuit ou paquet.

Le mode circuit permet de respecter l'ensemble des contraintes car une ressource est dédiée à un seul abonné. Malheureusement, ce mode de fonctionnement n'optimise pas l'utilisation des ressources. Prenons par exemple le cas d'une communication voix GSM : des études ont montré que les périodes de dialogue ne représentaient que la moitié de la durée de la communication. La ressource est donc fortement sous-utilisée.

C'est une des raisons majeures de l'introduction du mode paquet dans les réseaux cellulaires, permettant ainsi de maximiser le taux d'utilisation des ressources. Dans ce cas, plusieurs abonnées vont se partager lors de leur communication une même ressource. De plus, ce mode permet d'offrir aux abonnés de nouveaux services tels que l'accès Web[3] ou le streaming vidéo.

Malheureusement, il y a une contrepartie : il faut maîtriser le multiplexage (mélange) des flux applicatifs. Si cette maîtrise n'est pas acquise, on se rend bien compte que les contraintes de délai et de débit associées à chaque application ne seront pas respectées.

Le challenge consiste donc, pour l'opérateur, à fournir, sur demande, n'importe lequel de ces services, n'importe où, n'importe quand et dans des conditions prédéfinies de délai, débit et taux de perte.

Pour le support de la qualité de service dans les réseaux UMTS il est fondamental d'établir de façon analogue à la gestion de trafic dans les réseaux routiers, une architecture cohérente, un ensemble de règles et des mécanismes au niveau réseau, administration de réseau et services. Au niveau normalisation, le travail sur la qualité de service en est à ses débuts : seuls les concepts sont établis (débit, délai... par application). Par contre, il y a une réelle volonté de la part des comités de normalisation pour traiter ce sujet en profondeur. L'évolution sera lente et semblable à celle observée en GPRS. En d'autres termes, la première version commerciale des équipements UMTS ne permettra pas à l'opérateur de maîtriser l'ensemble des paramètres de qualité de service. Il faudra attendre l'horizon 2008.

Mais la maîtrise des paramètres de qualité de service au niveau réseau n'est pas suffisante. Indépendamment des équipements, l'opérateur UMTS doit dès à présent penser à introduire dans son système d'administration de réseau des outils de gestion de qualité de service.

Avec l'UMTS, le mobile ne sera pas seulement un téléphone portable mais également une radio, une télévision, un ordinateur avec une vitesse d'accès à Internet largement supérieure à celle des

modems analogiques téléphoniques classiques (modem 56ko). L'atout d'un mobile UMTS sera d'être l'unique support de l'ensemble de ces services multimédia en situation de mobilité.

Dans ce contexte, la qualité de service ne se limite peut être pas uniquement aux notions de débit et de délai mais aussi et surtout à la fourniture de contenus attractifs, c'est à dire de qualité.

2.3.4 La Qualité des Service dans le monde IP (Internet Protocol)

L'Internet est un réseau d'interconnexion de réseaux, s'appuyant sur divers moyens de transport, avec une grande variation de capacité de bande passante et de caractéristiques de délai. La nature sporadique des flux transportés conduit à de fortes variations de charge ; l'état des liens entre deux hôtes peut ainsi varier d'une milliseconde à l'autre. Plusieurs applications qui se partagent un lien peuvent, par instant, conduire à la congestion des ressources du réseau. La congestion génère délai de livraison et perte d'information.

[Shenker] définit « la qualité du service (QoS) » comme suit:

Quality of service refers to the nature of the packet delivery service provided, as described by parameters such as achieved bandwidth, packet delay, and packet loss rates. Traditionally, the Internet has offered a single quality of service, best-effort delivery, with available bandwidth and delay characteristics dependent on instantaneous load. Control over the quality of service seen by applications is exercised by adequate provisioning of the network infrastructure.

In contrast, a network with dynamically controllable quality of service allows individual application sessions to request network packet delivery characteristics according to their perceived needs, and may provide different qualities of service to different applications. It should be understood that there is a range of useful possibilities between the two endpoints of providing no dynamic QoS control at all and providing extremely precise and accurate control of QoS parameters.[4]

Jusqu'à présent, la solution la plus évidente pour éviter les périodes de congestion était de surdimensionner le réseau, avec pour objectif d'avoir un surplus de bande passante pour anticiper les périodes de haute demande. Cette solution n'est plus économiquement viable, car ces périodes

---

[3] La Qualité de service sur IP sera traité dans un chapitre à part.

[4] Au long du document, nous avons gardé certaines définitions en anglais. Ce choix a été fait à la suite de la non-existence d'une traduction officielle en français.

de forte charge sont impossibles à anticiper. Cette approche conduit de plus à une mauvaise utilisation des ressources, de grandes quantités de bande passante restant sans utilisation. La baisse vertigineuse du prix de la bande passante ne devrait pas suffire à satisfaire les besoins de toutes les applications sous toutes les conditions d'usage.

Si l'utilisation de l'IP ne peut aujourd'hui être remise en question, il est néanmoins nécessaire d'étendre l'architecture Internet afin d'élargir l'éventail d'offres de services (intégrant notamment la QoS). Cette évolution des réseaux IP nécessite l'introduction d'une intelligence capable de différencier le trafic puis d'offrir différents niveaux de service à chaque classe de service identifiée. Les réseaux IP ont donc notamment besoin d'une gestion de bande passante active. [Réf. INFRES].

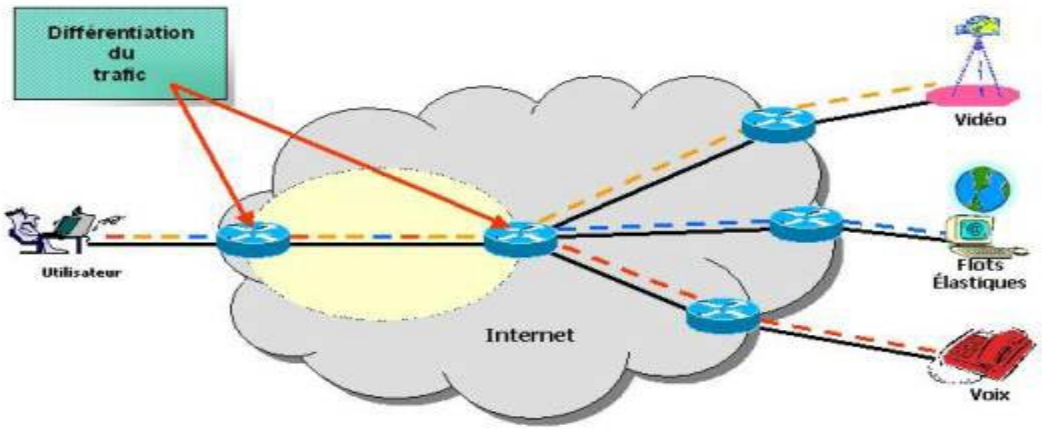


Figure 2-4 Différentiation du trafic sur un réseau IP

Les deux solutions qui ont été proposées au sein de l'IETF (Internet Engineering Task Force) sont :

- ***Réservation de Ressources (Modèle Intserv)***

Les ressources du réseau sont reparties par rapport à des demandes explicites des applications. La demande est satisfaite et une réservation est effectuée si suffisamment de ressources sont

disponibles. RSVP[5] est utilisé dans ce contexte comme protocole de signalisation.

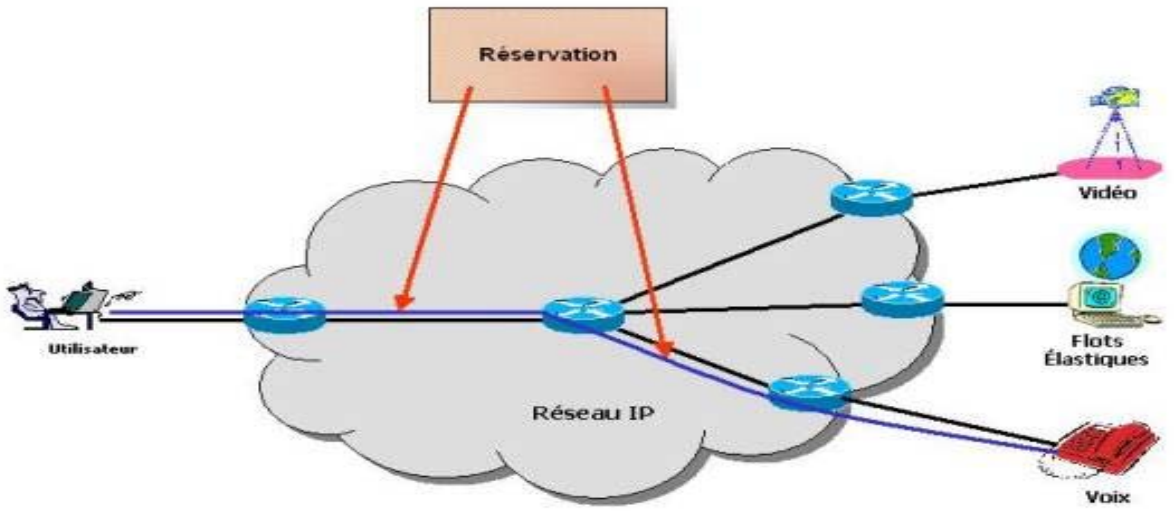


Figure 2-5 Modèle Intserv pour les réseaux IP

- ***Différentiation par Priorités (Modèle Diffserv)***

Le trafic entrant dans le réseau est classifié et des ressources lui sont attribuées, en fonction des critères de gestion du modèle de service. Aucune réservation n'est faite, mais un dimensionnement adéquat assure qu'il y aura assez de ressources dans le réseau pour les demandes de toutes les applications. Les garanties données par le modèle vont dans le sens du partage des ressources disponibles. Pour offrir un certain niveau de QoS, les classifications donnent un traitement différentiel à des applications sensées avoir des besoins plus exigeants.

[5] RSVP (Resource Reservation Protocol) is a set of communication rules that allows channels or paths on the Internet to be reserved for the **multicast** (one source to many receivers) transmission of video and other high-**bandwidth** messages.

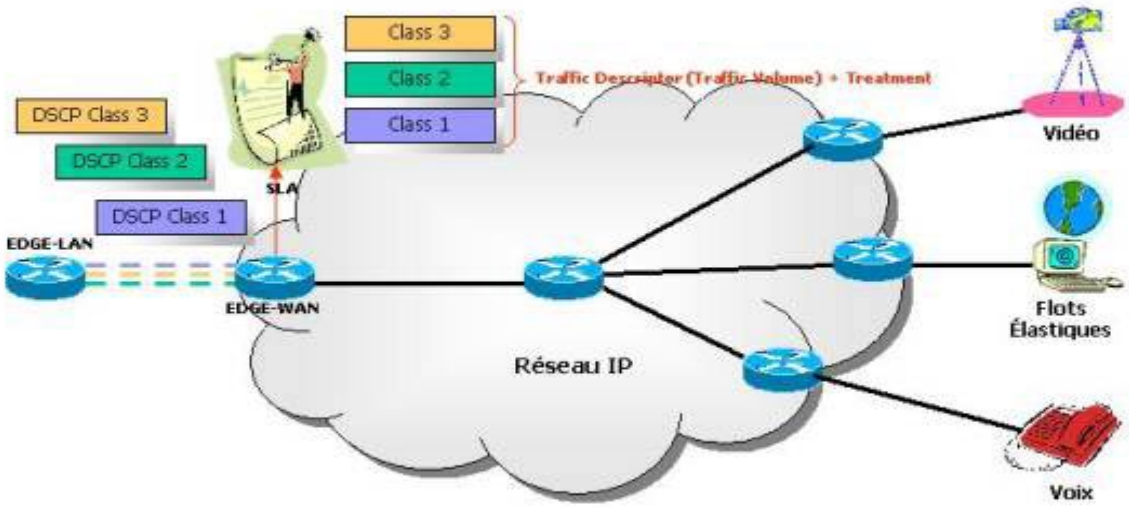


Figure 2-6 Modèle Diffserv pour les réseaux IP

Les deux solutions proposées ne sont pas mutuellement exclusives ou concurrentes, mais bien au contraire, elles sont complémentaires.

Nous allons présenter par la suite un chapitre sur la carte SIM du terminal mobile. Cette carte n'est pas directement liée à la qualité de service, mais tenant compte de la suite de nos travaux et la possibilité d'implémentation d'un outil de test pour la qualité de service, une introduction dans le domaine nous semble necessaire.

2.3.5 La carte SIM et le SIM ToolKIT

L'affectation d'un terminal à un client est un problème qui peut être résolu de différentes façons. La première, qualifiée de statique, consiste à introduire cette information dans une base de donnée. La seconde, dite dynamique, consiste à pouvoir modifier la possession du terminal depuis le terminal. C'est le rôle que joue la carte SIM dans le système GSM et sûrement dans les terminaux de la troisième génération.

Les cartes SIM font partie de la famille des cartes intelligentes (ou Smart Cards). Ce sont des cartes en plastique de dimension donnée incluant un semi-conducteur. Cet élément permet de stocker et souvent de traiter des informations. On distingue deux types fondamentaux de cartes à puce : les cartes à mémoire et les cartes à microprocesseur.

Cartes à mémoire

Les cartes à mémoire ne disposent pas de capacité de traitement et permettent simplement de stocker des informations. A ce titre, elles sont très proches des cartes à piste magnétique, elles présentent une capacité mémoire supérieure mais avecles mêmes faiblesses sécuritaires. Les cartes à mémoire sont destinées à des applications basiques telles que le débit d'unités téléphoniques.

Cartes à microprocesseur

Les cartes à microprocesseur incluent pour leur part une forte capacité mémoire et un microprocesseur leur conférant la possibilité non seulement de stocker des informations mais également de procéder localement à des traitements sur des données et à des calculs complexes d'algorithmes. Une des applications les plus évidentes des cartes à microprocesseur consiste à chiffrer les informations échangées par la carte et à mettre en oeuvre des mécanismes évolués de sécurisation pour contrer les tentatives de fraude. Ces cartes peuvent être utilisées comme cartes débit/crédit. Ces cartes à microprocesseur peuvent ainsi être considérées comme un petit ordinateur portable placé sur une plaque de plastique et capable de supporter des applications présentant un niveau de service évolué et une garantie de sécurité élevée vis à vis des utilisateurs. Ce concept sera utilisé pour l'UMTS.

2.3.5.1 La carte SIM du GSM

La carte SIM (Subscriber Identity Module ou Module d'Identification d'Abonné) est une carte à puce qui a la taille d'un timbre-poste et qui permet de localiser et de facturer l'utilisateur. Elle est indépendante du terminal et se glisse dans le lecteur de cartes dépendant ou indépendant du combiné GSM. L'utilisateur peut transférer sans difficulté ses abonnements d'un poste à l'autre et n'a donc plus besoin de se déplacer avec son terminal. Il lui suffit de glisser cette carte dans un terminal GSM pour qu'il soit activé. Le terminal reconnaît l'abonné grâce au code PIN (Personal Identification Number, code d'accès de la carte SIM).

Chaque abonné reçoit sa carte SIM qui regroupe un ensemble d'informations qui lui sont strictement personnelles :

- son code d'accès au réseau ;
- son type d'abonnement ;
- son répertoire principal.

La carte SIM procure des avantages à l'utilisateur et à l'exploitant. Elle apporte la sécurité par authentification de l'abonné, protection par code secret d'accès et confidentialté de la communication par codage. L'exploitant est capable d'établir une facturation simple et fiable.

Elle facilite la mobilité, car la localisation de l'abonné est indépendante du terminal. La mémorisation d'informations liées à l'abonné (numéros abrégés, confort des services) est assurée par carte SIM et facilite l'ergonomie du service. La carte est aussi conçue pour favoriser la mobilité sur des opérateurs différents.

2.3.5.2 Les évolutions des cartes SIM

Les cartes SIM sont un élément permettant la définition de services à valeur ajoutée (VAS). Plusieurs développements techniques permettent ces VAS : la taille de la mémoire et les outils de programmation.

- *La taille de la mémoire.*

La mémoire utilisée, appelée aussi EEPROM (Electrically Erasable Programmable Read Only Memory), est une mémoire programmable non volatile qui permet de stocker des applications. On assiste au doublement de la taille de stockage tous les deux ans (de 8Kbit à 32Kbits bientôt) ainsi qu'une évolution de la taille des mots.

- *Les outils de programmation.*

Le SIM ToolKit (STK ou SAT pour SIM ToolKit Application) est un standard arrêté par l'ETSI début 1998 qui permet à l'usager d'avoir accès aux VAS pour mobiles au travers de services accessibles par menus ajoutés aux menus traditionnels du terminal. Les menus STK et leurs services afférents sont logés sur la carte SIM. Ces STK se sont peu à peu répandus et font l'objet d'offres de la plupart des fournisseurs de cartes. Elles permettent d'imaginer des services évolutifs (voir Tableau 2-4). Deux facteurs ont favorisé la diffusion de cette technologie : une plus grande gamme

de terminaux compatibles avec les STK et l'existence de packages de conception de logiciels conçus par les fournisseurs de cartes à destination des opérateurs, leur permettant de concevoir leurs propres applications VAS en interne et de les tester rapidement. Cette nouvelle accessibilité a permis le développement d'offres marketing pour l'instant surtout basées sur l'information à la demande et les applications bancaires.

La première génération de STK est récente, mais la deuxième commence déjà à émerger. Une application STK de base donne à l'utilisateur un accès à un ensemble supplémentaire de menus sur son terminal. A chaque menu correspond un service. Dans la première génération de STK, ces menus et services sont gérés sur mesure en fonction de l'utilisateur, en modifiant des paramètres de la carte SIM à distance (over-the-air). Cette personnalisation des services et leur facilité de mise en oeuvre devraient permettre un large développement des services offerts. Les STK ne définissent que les commandes entre le mobile et la carte SIM, pas le langage utilisé.

Le langage JAVA est originellement défini comme un langage permettant de relier des équipements incluant un microprocesseur. Les fournisseurs de cartes SIM y ont vu l'intérêt d'établir un standard sécurisant les procédures de chargement des applets (certification d'applications). En collaboration au sein du Java Forum, ils visent à établir plusieurs standards successifs permettant de faciliter l'utilisation et la programmation des cartes à microprocesseur. Le but est d'éviter les systèmes propriétaires et de permettre à terme, à toute carte SIM d'accepter tout applet issu d'un environnement de cartes à microprocesseur (et donc de s'ouvrir à des applications non souscrites au départ, de recevoir tout type d'information ou d'échanges).

2.4 Etat de l’art des principaux travaux effectués

Afin de pouvoir placer l’utilisateur final dans « le monde de la QoS », il est important de passer en revue les initiatives académiques et industrielles en terme de qualité de service ainsi que les normes internationales, en particulier :

- Les normes ISO ;
- Les normes ITU-T ;
- Les normes ETSI.

C’est le sujet du paragraphe 2.4.

Au cours des dernières années, nous avons assisté dans différents secteurs au développement du travail d'encadrement de la Qualité de service focalisé sur les questions des utilisateurs, de la qualité de bout en bout, ainsi que des performances des réseaux. Celles-ci ont été conçues à l'origine par les opérateurs des télécommunications afin de mieux contrôler et gérer leur système de communication. La plupart des outils de mesure que nous trouvons sur le marché se focalisent sur la partie « réseau », « accès » ou « serveur » et très peu sur l'utilisateur final et ses besoins. Les projets académiques sont de plus en plus nombreux, mais leur intégration sur le marché est assez faible.

Il est impossible d'énumérer toutes les recherches menées dans le cadre académique et professionnel, mais les exemples suivants sont tout à fait représentatifs de l'état de l'art.

- *XRM Modèle de référence intégré et étendu (Extended Integrated Reference Model XRM)* par l'Université de Colombie, New York [Réf. XRM94]

XRM est un cadre pour la commande et la gestion des réseaux de télécommunications multimédia qui comportent des plates-formes de test multimédia pour des réseaux large bande. Le XRM est basé sur le travail théorique de garantie de la QoS dans des réseaux ATM et des systèmes qui contiennent des applications multimédia. Le modèle XRM peut être décrit comme une matrice (Tableau 2.5 Le modèle XRM) contenant 5 plans[6] (**N**etwork, **M**anagement, **D**ata, **C**ontrol et **U**ser) et 3 sous modèles différents (**R**ed, **G**reen et **B**lue). Une description plus détaillée est donnée en [Réf Columbia].

[6] Les plans correspondent à des differents niveaux dans l'architecture du modèle XRM

	R-model		**G-model**		**B-model**
*The **N-plane** (network & system management)*	*monitoring **individual** states, controlling **individual** states*		*monitoring **planes**, controlling **planes***		*monitoring **services**, controlling **services**, auditing, billing*
*The **M-plane** (resource control)*	*resource control (buffer and memory management, flow control, link and OS scheduling)*		***synchronization**, resource control (routing, admission control)*		*control of **services***
*The **D-plane** (data abstraction and management)*	*global distributed memory, network objects & media processor objects **(MIB)***	=======>	*Binding Interface Base **(BIB)**, directories, naming*	=======>	*MM mail, CSCW[7], Information gateways, VOD[8], PVM[9]*
*The **C-plane** (connection management and binding)*	***Signaling** (supports, e.g. distributed scheduling)*		*Stream control, Connection management, Binding*		*Service creation tools, navigation tools*
*The **U-plane** (user information transport and computing)*	*AAA protocol*		*Media stream protocol*		*Application protocol*
	Broadband Network & Media Processors	=======> *QoS Abstractions*	***Multimedia Network***	=======> *Service Abstractions*	***Service & Applications Network***

Tableau 2-5 Le modèle XRM

[7] **CSCW** : (Computer-Supported Cooperative Work)
[8] **VOD** : Video on Demande
[9] **PVM** : Parallel Virtual Machine

- *Architecture OMEGA* par l'Université de Pennsylvanie, Philadelphie [Réf.Nahrstedt95]

OMEGA examine le rapport entre les conditions de l'application de la QoS et la capacité de gestion de ressource locale et globale à satisfaire ces demandes. L'essence de l'architecture proposée est la réservation des ressources et la gestion des ressources de bout en bout. Pour faciliter ce processus de gestion de réservation de ressource, un *courtier* de QoS a été développé, qui incorpore la traduction de QoS, la négociation et la renégociation de la QoS.

- *Architecture MASI de bout en bout* par le laboratoire MASI, Université Pierre et Marie Curie, Paris [MASI94]

Cette architecture pour les communications a comme premier objectif le support de la QoS de bout en bout. L'architecture MASI offre un cadre générique de la QoS pour définir et mettre en place des niveaux de la QoS, des applications multimédias réparties sur des réseaux ATM. La gestion des ressources de bout en bout est incluse dans le logiciel d'exploitation du serveur central, le sous-ensemble du serveur et les réseaux ATM.

- *Architecture de la qualité de service sur IP* [CIS]

Ce site présente un nombre important de références concernant la qualité de service sur IP. Les références sont classées en plusieurs catégories : des articles, des livres concernant la QoS sur IP, des pages web, le travail de l'IETF concernant la QoS, des articles et des informations concernant le routage QoS, sur Diffserv et Intserv (voir section 2.3.4).

- *Architecture ARMOR* par l'INRIA, France [Réf. INRIA]

L'objectif central du projet ARMOR est l'identification, la conception et la sélection des architectures les plus appropriées pour la mise en place d'un service de communication, ainsi que le développement d'outils (informatiques, mathématiques) pour la réalisation de ces tâches. Cet objectif est abordé sous deux angles complémentaires : les aspects qualitatifs des systèmes (architectures, protocoles, procédures de contrôle) et les aspects quantitatifs, indispensables pour dimensionner correctement ces architectures et ces services (évaluation des performances, de la sûreté de fonctionnement, de la qualité de service, de la vulnérabilité, de la performabilité).

- *TINA (Telecommunication Information Networking Architecture)* par le consortium TINA

Le consortium industriel TINA regroupe plusieurs des principales entreprises de télécommunication du monde. Il indique des aspects de la QoS au niveau des applications distribuées à travers des architectures de réseaux d'ordinateurs. Son principal objectif est l'interopérabilité. De futurs réseaux de télécommunication devraient être développés conformément aux exigences de TINA.

On peut citer des programmes industriels et nationaux en faveur de la qualité tels que le Prix de la qualité Malcolm Baldridge ([MBNQAI]), le *Total Quality Management* ([TQM]) , *le QOSMIC (Quality of Service Verification Methodology and Tools for Integrated Broadband Communications)* ([QOSMIC]) , le TOMQAT (Total Management of Service Quality for Multimedia Applications ([TOMQAT]) ou encore le *Business Process Re-engineering* ([BPR]).

En matière de mesure de la qualité de service, nous trouvons actuellement sur le marché deux types de produits : les solutions prêtes à l'emploi et les boîtes à outils. Si dans la première catégorie les indicateurs de la QoS sont prédéfinis et codés dans le corps du logiciel, dans la deuxième, les utilisateurs ont la possibilité de construire leurs propres indicateurs à partir des variables des MIB[10] standard ou propriétaires. Parmi les sociétés qui proposent des produits de mesure que nous trouvons aujourd'hui, nous avons retenu :

- *Concord, Network Health*

L'offre Concorde comporte les composants suivants :

System Health est un gestionnaire d'alertes en temps réel ;

Application Health assure la disponibilité et les performances des applications ;

Network Health est la solution de supervision des performances de système d'information.

- *Infovista, Vistaview*

Infovista est une société française qui propose une offre complète appelée « *VistaView* » avec des rapports prêts à l'emploi, grâce auxquels une solution de mesure de niveau de service sur l'ensemble du système d'information peut être mis en œuvre.

- *IT Masters, Mastercell*

[10] MIB : Management Information Base – ce sujet sera traité plus en détails dans un chapitre à part.

Mastercell par son architecture orientée objet permet une bonne modélisation des ressources de systèmes d'information de l'entreprise. C'est un outil qui peut être complémentaire avec les autres outils précédemment décrits sachant qu'il est dirigé événement.

- *Lucent Software, VitalSuite*

La famille des produits VitalSuite comporte quatre composants principaux : VitalNet (métrique orienté réseau), VitalAgent (agent de supervision d'un poste utilisateur), VitalAnalysis (métriques et tableaux de bord pour les performances réseaux) et VitalHelp (détection des incidents techniques).

- *Quallaby, Proviso*

La gamme des produits Proviso est actuellement l'une des solution de suivi de performances du réseau et de la qualité de service les plus flexibles et le plus évolutives du marché. Proviso collecte, agence, archive et affiche les données du réseau pour les services à large bande.

- *Perform, Dashboard*

Dashboard est un outil qui propose des rapports de mesure détaillés pour évaluer la santé du réseau (routeurs, commutateurs, LAN, WAN et relais de trame). Par rapport à Proviso, il dispose d'un module auto-découverte. Une fois installé, le logiciel découvre les élements qu'il est capable de prendre en charge et démarre automatiquement.

La totalité des projets académiques et des produits industriels que nous trouvons sur le marché sont basés sur des normes internationales qui feront le sujet de notre prochain paragraphe.

2.5 Les normes internationales

2.5.1 Le rôle des normes dans les environnements distribués

Pour qu'une opération soit réalisée avec une parfaite interopérabilité entre les différentes applications multimédias distribuées, la terminologie, les modèles et les mécanismes doivent être très clairement définis. L'utilisation des normes internationales comme base de départ pour tout projet permet de mieux garantir les résultats obtenus.

La référence [Bradner] caractérise par exemple, *la norme Internet* comme suit:

« *A specification for which significant implementation and successful operational experience has been obtained may be elevated to the Internet Standard level. An Internet Standard (which may simply be referred to as a Standard) is characterized by a high degree of technical maturity and by a generally held belief that the specified protocol or service provides significant benefit to the Internet community. In general, an Internet Standard is a specification that is stable and well-understood, is technically competent, has multiple, independent, and interoperable implementations with substantial operational experience, enjoys significant public support, and is recognizably useful in some or all parts of the Internet.* »

Les normes ne peuvent pas tenir compte de toutes les idées académiquement appropriées. Cependant, toutes les initiatives mentionnées dans le domaine de la qualité de service (biens que d'autres domaines dans le monde des télécoms) démarrent le plus souvent sous la forme d'activités "non standardisées" académiques et, lorsque elles quittent leur environnement académique, s'orientent vers une standardisation internationale. Une fois que les résultats sont publiés, elles possèdent un certain niveau de maturité et en même temps, ils peuvent avoir à long terme une influence sur les fournitures au niveau de la standardisation.

La standardisation est devenue un processus ouvert. La coopération, la contribution et l'adoption sont explicitement encouragés dans les documents respectifs (voir par exemple, IETF [Réf. Brett]). Bien que les normes puissent à un certain moment être dépassées par les innovations techniques, leur remise à jour et leur révision périodiques peuvent leur permettre de rester opérationnelles.

Au niveau de la qualité de service, l'évolution des normes devient de plus en plus rapide. Notre recherche qui a été menée tout au long de ces trois dernières années, s'appuie sur les normes publiées par les organismes internationaux. Cependant, l'apparition des nouvelle normes ou des mises à jour au cours de ces dernières mois, n'ont pas pu être traitées et intégrées dans cette recherche (elles resteront une partie du travail futur).

2.5.2 Le choix des critères pour une comparaison

Afin de comparer les normes ISO, ITU-T et ETSI qui traitent l'évaluation de la QoS, nous appliquons les critères définis dans le Tableau 2-6. Ces critères ont été choisis par rapport à l'utilisateur final et sa perception au niveau de l'évaluation de la QoS.

Critère	Signification du Critère
L'Orientation de la norme	
Prise en compte de l'utilisateur final	Les points de vue et les soucis des utilisateurs finaux sont-ils pris en compte ?
Evaluation versus gestion	L'évaluation de la QoS est-elle considérée plutôt comme une gestion de la QoS ?
Orientation de l'application	La structure est orientée plutôt application-service ou plutôt réseau ?
Méthode d'évaluation	La procédure/méthode d'évaluation de la Qualité de service fait-elle partie de la norme ?
Les Caractéristiques de la Qualité	
La perception de l'utilisateur final	Les propriétés de la qualité telles qu'elles ont été définies sont-elles observées par les utilisateurs ?
Focalisation sur une partie des caractéristiques	Existe-t-il une sélection de certaines propriétés de la qualité ?
Les Propriétés de la Méthode d'Evaluation	
L'apprentissage	La méthode est-elle capable de former, d'apprendre l'utilisateur final?
L'autonomie	La méthode est-elle autonome ?
L'automatisation	La méthode peut-elle être automatisée ?
La facilité d'utilisation	La méthode utilise-t-elle des interfaces simples avec l'utilisateur ?
Les Résultats de l'Evaluation	
Discriminants	Est-ce que la mesure permet de déterminer le service le plus performant ?
Fidèles	Est-ce que la mesure va donner un résultat identique quelles que soient les circonstances si les paramètres du réseau sont identiques ?
Reproductibles	Les résultats obtenus sont-ils reproductibles ?

Tableau 2-6 Choix des critères

2.5.3 Les normes ISO

ISO (International Organisation for Standardisation) [ISO00], dispose de deux normes autonomes spécifiques à la qualité de service: ISO 9000 et ISO 9126.

2.5.3.1 ISO 9000

Les discussions sur la qualité aboutissent très souvent sur les normes les plus connues, notamment la norme ISO 9000. Cette norme convient pour n'importe quel modèle de la qualité de service. Reste à savoir si elle est applicable aux services multimédia et adaptée à des applications dans les environnements distribués.

En 1987, les normes internationales ISO 9000 (de ISO 9000 à ISO 9004) sont sorties dans leur première version. Elles constituaient alors des normes génériques pour la gestion et la garantie de la qualité. Elles contiennent des indications précieuses pour tout fournisseur de produits, désireux de mettre en application des systèmes de qualité efficaces au sein de son entreprise ou d'améliorer les systèmes de qualité existants (gestion de qualité). Elles comportent aussi les conditions génériques permettant au client d'évaluer le système de qualité d'un fournisseur (garantie de la qualité).

Trois d'entre elles sont des **normes de conformité**. Elles portent sur les systèmes de qualité et sont employées pour la garantie externe de la qualité. Les trois modèles alternatifs représentent les formes distinctes "de possibilités fonctionnelles ou d'organisation" propres aux deux parties contractantes [IS09000] :

- Les Systèmes de la Qualité – Système qualité - Modèle pour l'assurance de la qualité en conception, design, développement, production, installation, et prestations associées [Réf. IS09001],

- Les Systèmes de la Qualité – Système qualité - Modèle pour l'assurance de la qualité en production, installation, et prestations associées [Réf. IS09002]

- Les Systèmes de la Qualité – Système qualité - Modèle pour l'assurance de la qualité en contrôle et essais finals [Réf. IS09003].

Un certificat attestant l'existence et l'efficacité d'un système de qualité se rapporte toujours à l'une de ces trois normes. En revanche, ISO 9000 et ISO 9004 sont dites " **normes de conseils**".

ISO 9004-2 est la norme la plus adaptée aux services et aux applications. Un " service *" est un résultat produit par des activités à l'interface entre le fournisseur et le client, et par des activités internes de fournisseur pour satisfaire les besoins d'un client* [Réf. IS08402]. Des services de télécommunication sont explicitement mentionnés.

La question principale d'ISO 9004-2 est le concept " D'une Boucle de qualité de service " avec trois processus : le marketing, le processus de la conception (design) et le processus de la livraison de service. Les plus importants sont le processus de conception et le processus de livraison de service. Des mesures de la Qualité de service effectuées à la fois par des clients et des fournisseurs sont utilisées pour commander le processus de la livraison de service et pour améliorer tous les processus impliqués.

Les résultats du processus de la conception sont les spécifications du service, les spécifications de la livraison de service et les spécifications du contrôle de qualité. Les spécifications de service définissent le service à fournir. Elles décrivent clairement les caractéristiques du service qui est en effet l'évaluation du client. Le fournisseur de service doit pouvoir faire aussi une évaluation à l'aide de caractéristiques comparables aux normes d'acceptabilité définies. Elles sont mesurables par le client ou par le fournisseur de service.

La conception de la livraison de service définit les moyens et les méthodes employés pour fournir le service. Elle décrit clairement les caractéristiques de la livraison de service qui ont un impact direct sur la qualité du service, y compris la définition des conditions des ressource, du nombre et du profil du personnel exigé et de la confiance dans des fournisseurs pour les produits et les services achetés. Les caractéristiques de la livraison de service ne peuvent pas toujours être observables par le client (perception d'utilisateur), mais il y a un besoin général d'évaluation par rapport aux normes définies de l'acceptabilité. Ceci doit être fait par le fournisseur de service, qui devrait aussi fournir les caractéristiques quantitatives de la livraison de service.

Le contrôle de la qualité doit être considéré comme partie intégrale du processus de service. Les spécifications de contrôle de la qualité définissent les procédures pour évaluer et commander les caractéristiques du service ainsi qie sa livraison. La conception du contrôle de la qualité implique d'une part d'identifier et d'analyser les activités principales dans chaque processus pour choisir les caractéristiques appropriées en ce qui concerne la qualité de service d'une part, et d'autre part de définir des méthodes pour évaluer et commander ces caractéristiques.

Les normes ISO 9000 ont été conçues « par les entreprises et pour les entreprises ». Les nouvelles versions (année 2000) seront intégrées comme des puissants outils de management, car les nouvelles dispositions introduites révèlent de réelles opportunités de progrès et d'amélioration des performances de la qualité. Leur nouveau recentrage sur la satisfaction des clients et le management de processus « permettront de réconcilier la qualité au service de la stratégie de l'entreprise » [Réf. AFNOR].

« Les organismes dépendent de leurs clients et il convient, par conséquent, qu'ils comprennent leurs besoins actuels et futurs, répondent à leurs exigences et s'éfforcent d'anticiper leurs attentes. L'organisme détermine les besoins et attentes des clients ainsi que les exigences imposées, puis les convertit en exigences spécifiées. » [Réf. ISO 9000 : 2000]

Un organisme ne travaille pas à la seule fin de produire pour produire, mais surtout pour satisfaire les besoins de ses clients et même d'anticiper leurs attentes. Le client est la raison d'être de l'organisme et se trouve donc impliqué dans tous ses processus.

Les normes ISO 9000 (année 2000) mettent l'accent sur la satisfaction des exigences des parties intéressées de l'organisme et notamment de celui (ou ceux) à qui elle fournit un produit ou un service, désigné ci-après par «le client» [Réf. ISO 9001].

Résumé d'ISO 9000

Les séries ISO 9000 de normes internationales définissent des conditions de la qualité du système. Elles fournissent des conseils pour établir un système de qualité. En raison de la grande variété de compagnies, de produits, et de services existants, les normes ISO 9000 laissent intentionnellement une large marge pour des réalisations spécifiques d'un système de qualité.

Fondamentalement, les normes ISO 9001, ISO 9002, et ISO 9003 exigent d'une entreprise de consulter ces normes, de passer en revue ces processus pour voir s'ils sont toujours conformes et éventuellement de faire des changements si nécessaire. Bien que l'évaluation des conditions de la qualité soit mentionnée, les méthodes concrètes de leur mise en place ne sont pas décrites.

Le problème principal avec les séries ISO 9000 est que ces standards de qualité sont liés davantage à la gestion de qualité qu'à la production de la qualité du service elle-même - ce qui est la question la plus importante par rapport à la perception des utilisateurs. Les standards dans les séries internationales des normes ISO 9000 ne sont pas des standards en ce qui concerne le produit. Elles

n'incluent aucune exigence technique : « *les exigences de la qualité du système sont complémentaires aussi, mais irremplaçables techniquement parlant pour des produits distincts.* »

2.5.3.2 ISO 9126

[IS09126] (Technologie de l'information - Evaluation des logiciels - Les caractéristiques de la qualité et les directives pour leur usage) est un cadre parmi les premiers (1991) et un modèle généralisé de qualité pour l'évaluation de la qualité du logiciel. Il adresse à la fois, la qualité du produit logiciel et la qualité du procédé de développement. Il vise six caractéristiques qui définissent, avec un chevauchement minimal, la qualité du logiciel : fonctionnalité, fiabilité, rentabilité, efficacité, entretien et portabilité.

Il est dit dans la norme *« que ces caractéristiques peuvent être applicables à chaque genre de logiciel, y compris des programmes machine et des données contenus dans les progiciels ».* Par conséquent, ISO 9126 est une bonne base pour l'évaluation de la qualité appliquée aux services, car les services eux-mêmes peuvent être compris comme parties de logiciel[11].

La norme actuelle ISO 9126 est en cours de révision. La nouvelle norme sera divisée en trois parties :

- la partie 1 [Réf. **I**SO/IEC 9126-1] couvrira les caractéristiques de qualité et les sous caractéristiques qui ont été déplacées de l'annexe dans la norme. Nous trouvons aussi une définition de la qualité qui est basée sur le point de vue de l'utilisateur comme résultat de l'usage d'un logiciel.

- la partie 2 adressera la métrique externe [Réf. ISO/IEC 9126-2] de la mesure de qualité du logiciel. La métrique externe est une échelle quantitative et une méthode qui peut être utilisée pour mesurer une caractéristique d'un logiciel dérivée du comportement du système, dont elle fait partie au moment de la mesure.

- la partie 3 traitera la métrique interne [Réf. ISO/IEC 9126-3] de la mesure de qualité du logiciel. La métrique interne est une échelle quantitative et une méthode qui peut être utilisée pour mesurer une caractéristique d'un logiciel indépendamment des autres caractéristiques du système au moment de la mesure.

- la partie 4 traitera la métrique de la qualité du logiciel utilisé [Réf. ISO/IEC 9126-4]. Cette métrique est traduite par l'efficacité, la productivité, la satisfaction et la sécurité du logiciel.

Résumé d'ISO 9126

ISO 9126 n'est pas approprié à notre objectif pour deux raisons fondamentales. Bien qu'elle comporte des solutions pour l'utilisateur, de méthodes d'évaluation, et la couche application, elle s'arrête trop tôt en ne décrivant pas les méthodes à employer, ni les résultats qui peuvent être atteints. ISO 9126 propose d'abord de faire la mesure de chaque caractéristique de la qualité. Puis l'estimation et l'évaluation doivent suivre. Du point de vue d'un utilisateur cette évaluation est trop tardive, compliquée et incompréhensible dans la mesure où les caractéristiques essentielles de la qualité ne ressortent pas de l'ensemble des caractéristiques mesurées.

2.5.4 Les Normes ITU-T

ITU-T, le secteur ITU de Télécommunication appelé précédemment CCITT, est un organe permanent de l'Union Internationale de Télécommunication (ITU) qui a la responsabilité d'étudier les questions techniques, le fonctionnement et la tarification pour les télécommunications mondiales [Réf. ITU-T00]. ITU-T/CCITT a édité des centaines de documents, recommandations, et normes pour chacune des sept couches de leur architecture de référence OSI (*Open Systems Interconnection*, ce qui se traduit par *Interconnexion de systèmes ouverts)*. Ces recommandations sont la base de beaucoup de services de télécommunication et des applications aujourd'hui.

2.5.4.1 I.350

La norme I.350 [Réf. I.350] illustre *comment la qualité du service et les performances du réseau sont prise en compte dans les réseaux numériques, y compris les RNIS*[12]. Cette norme a comme point de départ la définition de la qualité de service dans la recommandation E.800 [Réf. E.800] : *effet collectif des performances de service qui déterminent le degré de satisfaction d'un usager du service*. La définition étant si vaste, les aspects de qualité de service qui sont traités dans la présente norme se limitent à l'identification des paramètres qui peuvent être observés directement et mesurés

[11] Nous emploierons ISO 9126 par exemple dans la section 4 " Evaluation de la qualité de service pour des services de courrier électronique "
[12] RNIS : Réseau Numérique à Intégration de Services

au point où l'usager peut accéder au service considéré. I.350 offre une bonne distinction entre la qualité du service et les performances du réseau (Tableau 2-7).

Qualité de service	Performance du réseau
Orientée usager	Orientée exploitant
Attribut du service	Attribut de l'élément de connexion
Centrée sur les effets observables par l'usager	Centrée sur la planification, le développement (conception), l'exploitation et la maintenance
Entre (aux) points d'accès au service	De bout en bout ou possibilités des éléments de connexion du réseau

Tableau 2-7 Distinction entre la qualité de service et les performances du réseau

I.350 distingue les notions de services supports et téléservices. En particulier, les paramètres qui décrivent la QOS d'un service support, ne sont pas du même type que ceux qui décrivent la QOS d'un téléservice. Dans le cas de téléservices, l'interface entre l'usager et le prestataire du service peut être une interface homme-machine. Dans le cas de services supports, cette interface correspond aux points de référence S (point de liaison entre 2 terminaisons de réseau) ou T (point de liaison entre une terminaison de réseau et un adaptateur de terminal).

En ce qui concerne les performances du réseau, I.350 présente « les paramètres de performance primaires » (observation directes d'événements aux points d'accès aux services ou aux limites d'éléments de connexion.) et « les paramètres de performances dérivés » (paramètre ou mesure de paramètre établi à partir de valeurs observées d'un ou de plusieurs paramètres de performance primaires pertinents et de seuils de décision pour chaque paramètre de performance primaire pertinent).

Neuf paramètres primaires génériques de performance du réseau ont été établis suivant la méthode matricielle. Nous pouvons les utiliser pour définir des paramètres QoS et performance du réseau spécifique:

– rapidité de l'accès ;

– précision de l'accès ;

– sûreté de fonctionnement de l'accès ;

– rapidité du transfert de l'information ;

– précision du transfert de l'information ;

– sûreté de fonctionnement du transfert de l'information ;

– rapidité du retrait[13];

– précision du retrait ;

– sûreté de fonctionnement du retrait.

La matrice (Figure 2-7) constitue une méthode systématique d'identification et d'organisation des éventuels paramètres de performance du réseau en vue de définir un ensemble concis de paramètres et, le cas échéant, de leurs contreparties: en termes de QoS.

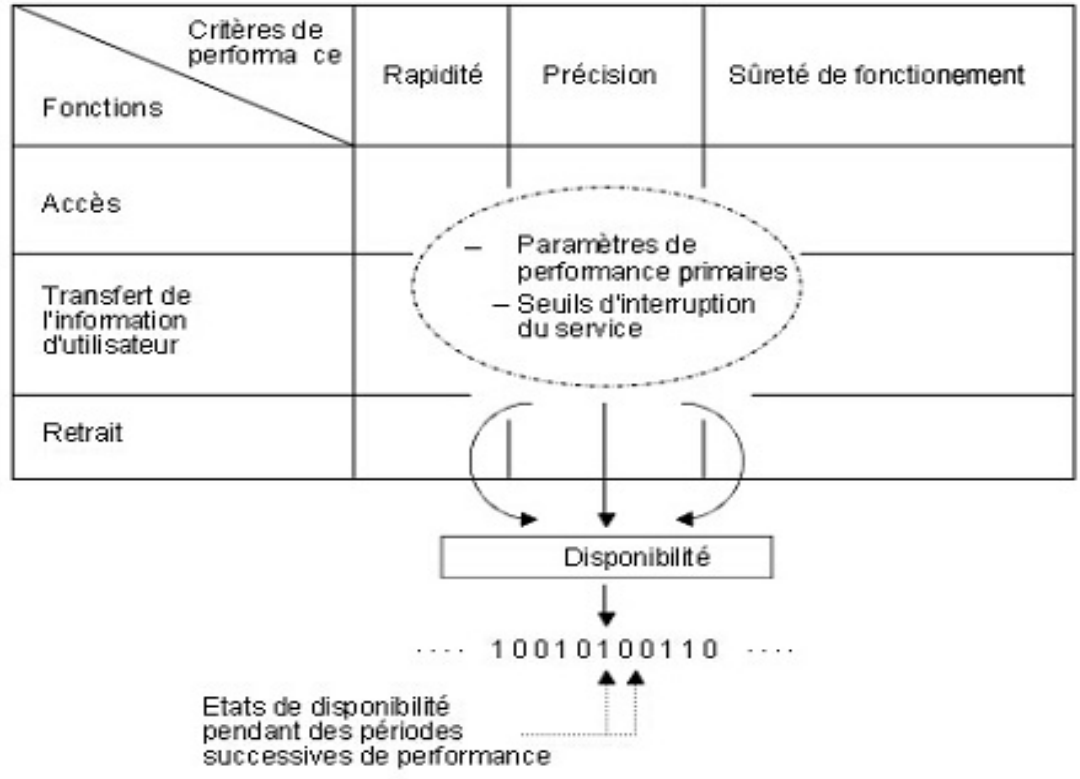


Figure 2-7 *Méthode systématique d'identification et d'organisation paramètres de performance du réseau*

[13] Le retrait englobe à la fois les activités de déconnexion du circuit physique (si nécessaire) et de terminaison du protocole de niveau supérieur.

Dans les annexes de la norme, on trouve 4 tableaux qui montrent les relations qualitatives entre les paramètres génériques primaires définis et un ensemble de paramètres QoS des services supports, ainsi que des performances du réseau pour le mode de commutation de circuits, pour le mode de commutation par paquets et pour le RNIS à large bande. Etant donné notre approche QoS, c'est le premier tableau (Tableau 2-8) qui sera présenté.

Paramètres génériques		**Paramètres de QOS pour les services supports**														
		Paramètres de performance primaires												*Paramètres de performance dérivés*		
		Temps d'accès	Probabilité d'accès incorrect	Probabilité de refus d'accès	Temps de transfert de l'information d'usager	Débit de transfert de l'information d'usager	Probabilité d'erreur dans l'information d'usager	Probabilité de remise d'une information d'usager excédentaire	Probabilité d'erreur dans la remise de l'information d'usager	Probabilité de perte de l'information d'usager	Temps de retrait	Probabilité de retrait incorrect	Probabilité de refus de retrait	Disponibilité du service	Probabilité de refus de transfert de l'information d'usager	Durée d'interruption du service
Primaires	Rapidité de l'accès	X														
	Précision de l'accès		X													
	Sûreté de fonctionnement de l'accès			X												
	Rapidité du transfert de l'information				X	X										
	Précision du transfert de l'information						X	X	X							
	Sûreté de fonctionnement du transfert de l'information									X						
	Rapidité du retrait										X					
	Précision du retrait											X				
	Sûreté de fonctionnement du retrait												X			
Dérivés	Disponibilité													X	X	X

Tableau 2-8 Relation qualitative entre les paramètres de performance génériques et les paramètres de QOS possibles pour les services supports

Résumé d'I.350

Si les paramètres concernant l'accès au service sont traités en détails, d'autres types de paramètres de QoS, qui sont de nature subjective, c'est-à-dire qu'ils dépendent soit d'actions de l'usager, soit de son opinion subjective, ne seront pas spécifiés dans les recommandations de la série I relatives à la QoS. La méthode décrite (" L'approche de la matrice 3x3 ") pour identifier des paramètres va dans le même sens, mais elle est assez compliquée et non adaptée à l'utilisateur final. Néanmoins l'application de I.350, une fois comprise et appliquée, peut fournir des résultats significatifs et reproductibles.

2.5.4.2 X.641 / X.642

La norme [X.641] [Réf. X.641] a été préparée en collaboration entre ISO, IEC, et ITU-T. Elle est identique au document d'ISO/IEC 13236. Son but *est de fournir une base commune pour le développement et le perfectionnement de l'éventail de normes qui indiquent ou référencent les conditions ou les mécanismes de la qualité de service (QoS) dans un environnement de la technologie de l'information (IT)*. Il est complémentaire aux recommandations existantes ITU-T, qui sont plus orientées vers les performances et la signalisation du réseau au niveau de la QoS. Si au début cette norme était censée compléter le modèle de référence de l'interconnexion de systèmes ouverts (OSI - Opening Systemes Interconnection), maintenant ce n'est plus le cas.

La cible pour X.641 est les concepteurs de normes nouvelles ou révisées, qui se rapportent aux technologies de l'information, qui définissent ou utilisent les mécanismes de qualité de service et les utilisateurs des technologies de l'information, pour exprimer leurs attentes en matière de qualité de service.

Dans ce rôle, [X.641] est un cadre conceptuel utile qui est basé sur des différents concepts :

- *le service* auquel la qualité de service s'applique ;

- *les caractéristiques* de qualité de service décrivant les aspects fondamentaux de la qualité de service qui doivent être gérés ;

- *les prescriptions et les politiques* en matière de qualité de service ainsi que les fonctions de gestion de la qualité de service permettant de les mettre en œuvre ;

- *les mécanismes* de qualité de service de base qui se conjuguent pour former les fonctions de gestion de la qualité de service ;

- *les catégories* de qualité de service correspondant aux différents ensembles de prescriptions des utilisateurs en matière de qualité de service imposées par certains environnements (les communications en temps réel, par exemple) ou par la politique au niveau des systèmes ;

- *les stades d'une activité* auxquels la qualité de service doit être gérée.

Le [X.641] se focalise sur certaines conditions de la QoS et dans une certaine mesure sur la gestion de la QoS. Pour avoir des caractéristiques détaillées pour des mécanismes de la QoS, il faut se focaliser sur une autre norme, la [X.642] [Réf X.642] qui a été préparée en collaboration entre ISO, le IEC, et ITU-T. Elle est identique au document d'ISO/IEC 13243 et utilise les concepts et la terminologie de X.641.

Pour mieux interpréter et comprendre l'utilité de ces normes, nous pouvons essayer de positionner l'utilisateur final et la possibilité d'évaluation de la qualité de service de sa part dans le cadre du modèle OSI.

Si nous suivons les mécanismes de fonctionnement du modèle, ainsi que les détails qui sont précisés dans les 2 normes (X.461 et X.462), nous nous apercevons que l'utilisateur final ne participe au processus d'évaluation de la QoS qu'à partir du moment où il décide d'utiliser un service. L'utilisation du service se traduit en fait par des caractéristiques techniques de la qualité de service. Les entités qui reçoivent ces caractéristiques les analysent et déterminent la fonction de gestion de la QoS ou les mécanismes qui sont exigés pour les réaliser. À partir de ce moment là, l'utilisateur est hors du jeu. Il n'y a aucune occasion ou moyen de suivre une évaluation de la qualité de service.

Résumé de X.641/X.642

X.641/X.642 est une autre approche pour définir chacune des caractéristiques d'un système, d'un service utilisé. La QoS *est indépendante de la façon dont la QoS est représentée ou commandée* dans un vrai système. Par sa nature générique, X.641 est large-trop large si on se concentre sur les quelques caractéristiques importantes du point de vue d'un utilisateur. Le cadre OSI pour la QoS définit un ensemble de caractéristiques de la QoS pour des systèmes de communication et de traitement. Ils sont très généraux et ont besoin d'amélioration pour être utiles, particulièrement pour l'évaluation.

2.5.4.3 X.900

X.901-X.904 (**Technologies de l'information – Traitement réparti ouvert – Modèle de référence**) est l'ensemble de normes ITU-T qui correspondent au document ISO/IEC 10746 concernant les applications réparties (voir section 1.1.2). La croissance rapide de ces applications *a fait naître le besoin d'un cadre pour coordonner la normalisation du traitement réparti ouvert (ODP[14]). Le Modèle de référence ODP fournit ce cadre. Il établit une architecture qui permet la prise en compte de la répartition, l'interfonctionnement et la portabilité.* [Réf. X.903]. Les normes reposent sur des concepts précis issus des développements récents dans le domaine des traitements répartis et s'appuient, dans la mesure du possible, sur l'utilisation des techniques de description formelle pour la spécification de l'architecture.

L'intérêt principal du modèle de référence d'un processus distribué ouvert (Reference Model of Open Distributed Processing RMODP) concerne l'aspect de la distribution dans les systèmes OSI. La cible peut inclure des créateurs de standards et des architectes des systèmes.

Le standard du processus distribué ouvert (Open Distributed Processing - ODP) couvre la modélisation de l'objet avec des spécifications du système, la définition d'une infrastructure du système, et finalement un cadre pour évaluer la conformité du système. La norme se compose de quatre parties majeures décrivant :

- le modèle dans une vue d'ensemble et une introduction sur les instructions [Réf. X.901] ;

- les concepts de base pour les systèmes répartis [Réf. X.902] ;

- l'architecture des systèmes répartis [Réf. X.903] ;

- la sémantique architecturale avec une formalisation des concepts de modélisation ODP définis dans la X.902.

Le modèle de référence d'un processus de distribution ouvert (Reference Model for Open Distributed Systems - RMODP) a été développé pour fournir une architecture générale pour les environnements distribués ouverts. Il définit en particulier :

– cinq *points de vue*, appelés entreprise, information, traitement, ingénierie et technologie, qui fournissent des éléments de base pour spécifier des systèmes ODP;

– un *langage de point de vue* pour chaque point de vue, définissant les concepts et règles de spécification des systèmes ODP pour le point de vue correspondant ;

– la spécification des *fonctions* requises par les systèmes ODP ;

– des *règles de transparence* précisant comment utiliser les fonctions ODP pour assurer une transparence à la répartition.

L'architecture des systèmes ODP et la composition des fonctions sont déterminées par la combinaison du langage de traitement, du langage d'ingénierie et des règles de transparence.

Les cinq points de vue sont des spécifications d'un système entier lié à un ensemble particulier des problématiques:

- **point de vue entreprise** : point de vue sur un système ODP et son environnement détaillant les objectifs, le domaine d'application et la description d'entreprise. A l'aide de cette norme, on peut élaborer des spécifications communes pour des caractéristiques particulières à un domaine industriel ou à des fonctions commerciales, comme des

[14] ODP : Open Distributed Process

politiques de sécurité, qu'il y a lieu de soutenir en termes de contrôle d'accès et de niveau de protection. Ces normes se rapportent au domaine commercial ou applicatif concerné. Ce point de vue peut être modélisé sous forme de contrats, agents ou rôles [Réf Blair].

- **point de vue information** : point de vue sur un système ODP et son environnement couvrant la sémantique de l'information et son traitement pour élaborer des spécifications communes pour les objets d'information. Il peut être modélisé sous forme d'objets, d'objets composés ou de schémas (statique, dynamique, invariable) [Réf Blair].

- **point de vue traitement** : point de vue sur un système ODP et son environnement permettant la prise en compte de la répartition grâce à la décomposition fonctionnelle du système en objets interagissant à leurs interfaces. Les spécifications des interfaces seraient définies en termes d'opérations abstraites et de types de données abstraites correspondant aux objets d'information de la description d'information. Les normes concernant les opérations abstraites pourraient être communes à plusieurs domaines d'application, celles concernant les types de données abstraites seraient définies pour le domaine d'application concerné. Ce point de vue peut être modélisé soit comme un objet, une interface, ou un contrat environnemental [Réf Blair].

- **point de vue ingénierie** : point de vue sur un système ODP et son environnement couvrant les mécanismes et les fonctions nécessaires pour mettre en œuvre les interactions réparties entre objets du système. Il peut être modélisé comme des bouts, des objets de protocole ou des nœuds [Réf Blair].

- **point de vue technologie** : point de vue sur un système ODP et son environnement couvrant les choix de technologie dans ce système. Il peut être modelisé comme une implémentation, des points de conformité ou Implementation eXtra Information for testing (IXIT) [Réf Blair].

Les points de vue entreprise, information, traitement, ingénierie et technologie ont été choisis comme constituant un ensemble nécessaire et suffisant pour répondre aux besoins des normes ODP.

Ces points de vue peuvent être liés aux contraintes de la qualité de service [Réf Leydekkers96]. La même référence émet des recommandations concernant la construction des systèmes répartis qui sont confrontés aux contraintes de la qualité de service à partir du X.900. Le Tableau 2-9 montre différents niveaux de la QoS inclus dans le modèle ODP.

Le point de vue ODP	**Le niveau de la QoS**	**Les caractéristiques**	**Les exemples**
L'entreprise	*QoS subjective*	Les besoins QoS orientés vers l'utilisateur final (non formel et subjective)	« L'audio doit être tout au long de la vidéo » « La vidéo doit être similaire à la télévision en termes de qualité »
L'information	*QoS objective*	Le rapport précis sur les besoins qui dérivent des spécifications subjectives de la QoS. Souvent il s'agit d'applications indépendantes.	25fps[15] (PAL), 30 fps (NTSC) format 4 :3 (format de PAL TV). Largeur de bande du téléphone 8 kHz
Le traitement	*QoS de l'Application*	Décrit les besoins en termes de QoS pour des applications multimédia	La synchronisation entre l'audio et la vidéo peut être exprimée en ms. (ex. 80ms)
L'ingénierie	*QoS Système / QoS du Réseau*	Décrit les besoins en terme de QoS pour le système d'exploitation. Décrit les besoins en terme de QoS pour le réseau	La taille du buffer, les opérations, la mémoire (en MB) Le débit de sortie (en Mbit/s), le jitter (en ms), le retard (en ms)
La technologie	*Propriétés de la QoS*	Décrit les caractéristiques de la QoS en terme de moyens du système d'exploitation et la technologie dans le réseau.	Le dispositif visuel (format PAL), le dispositif audio (μ-law), taux de perte pour l'ATM

Tableau 2-9. Différentes niveaux de QoS liés au point de vue ODP

On peut observer que le RMODP est déjà une convention utilisée pour décrire des besoins de QoS des 5 points de vue différents et que les questions de la QoS sont bien reflétées dans ce cadre. L'intérêt principal pour un utilisateur est représenté par l'entreprise, l'information et le traitement.

[15] fps : frame per second (nombre de trames par seconde)

Dans la norme X.902, nous trouverons les notions d'un contrat d'environnement liant un objet et son environnement, incluant des contraintes de qualité de service, d'usage et de gestion. Les contraintes de la qualité de service comprennent des contraintes temporelles (par exemple des dates limites), des contraintes de volume (par exemple des contraintes de débit) et des contraintes de sûreté de fonctionnement couvrant des aspects liés à la disponibilité, la fiabilité, les facilités de maintenance, la sécurité (par exemple le temps moyen entre deux défaillances). Les contraintes d'usage et de gestion comprennent les contraintes de position (c'est-à-dire les positions dans le temps et dans l'espace qui sont choisies) et les contraintes de transparence à la répartition (c'est-à-dire les transparences à la répartition sélectionnées). Les contraintes d'environnement peuvent décrire à la fois des exigences concernant l'environnement d'un objet pour garantir un comportement correct de l'objet et des contraintes sur le comportement d'un objet dans un environnement correct. Ces contraintes peuvent être incluses dans les Service Level Agreement.

Résumé X.900

X.900 est un cadre très puissant qui fournit des mécanismes pour décrire des conditions de la QoS. Il oublie cependant le besoin des utilisateurs d'évaluation de la qualité de service. Dans X.900 ni les interfaces d'évaluation ni des outils de coopération avec des systèmes, d'après ODP, ne sont indiqués. Indépendamment de cela, ODP correspond à des spécifications fortement abstraites, qui le rendent très difficile à comprendre par les utilisateurs. Des extensions qui traitent l'évaluation de la QoS, seront possibles à l'avenir.

2.5.5 Les Normes ETSI

L'ETSI "European Telecommunications Standards Institute" [Réf. ETSI] représente une association non lucrative localisée au sud de la France à Sophia Antipolis, dont le rôle principal est de promouvoir les télécommunications au sein de l'Europe et de contribuer à la réalisation des normes internationales. Ses 912 membres venant de 54 pays sont des opérateurs, des constructeurs, des fournisseurs de service, des chercheurs et des utilisateurs. Les travaux effectués au sein du groupe de travail « User Group» et plus précisément le document EG 202 009 « Quality of Telecom Services » [Réf. EG 202 009], nous intéresse tout particulièrement.

2.5.5.1 EG 202 009

La Qualité dans le domaine des services de télécommunication peut être évaluée sous différents points de vue avec différentes méthodes :

- un premier niveau de QoS peut être relié à la fiabilité des équipements et peut être évalué via des méthodes techniques assez coûteuses à cause du nombre important de mesures et de l'incompatibilité des différents résultats par rapport à des équipements différents ;

- un second niveau peut être relié à la disposition du service qui est lié à l'usage de ce service. Pour cette raison, des critères appropriés doivent être définis par rapport au type d'usage entre le client et le fournisseur ;

- un dernier niveau qui est censé mesurer la satisfaction subjective du client.

Si pour les deux premiers niveaux les standards existent, pour le dernier les travaux manquent. C'est la raison pour laquelle, cette première version donne une méthodologie pour analyser les besoins des utilisateurs qui est le premier pas dans le processus du Management Total de la Qualité de Service. (Total Quality Management – TQM).

Le document est divisé en trois parties :

- Une méthodologie pour l'identification des paramètres significatifs pour l'utilisateur ;

- Les paramètres apparentés par rapport à un service spécifique ;

- Un corpus pour établir un SLA (Service Level Agreement).

Plusieurs types de qualité de service sont présentés :

- *la QoS offerte par le fournisseur de service* qui est exprimée par des valeurs spécifiques à chaque service proposé. Ces paramètres doivent être compréhensibles par les utilisateurs ;

- *la QoS accomplie par le fournisseur de service* qui est exprimée par des valeurs affectées aux paramètres QoS qui ne doivent pas être éloignés des paramètres de la QoS offerte ;
- *la QoS perçue par l'utilisateur/le client* qui est exprimée normalement en termes de degré de satisfaction et non pas en termes techniques. Les termes techniques peuvent être présents mais uniquement quand le client est capable de les comprendre et de les utiliser ;
- *la QoS en termes d'exigences des utilisateurs* qui sont exprimées dans un langage non technique et qui doivent être traduites par les fournisseurs de service en indicateurs techniques faciles à mettre en place.

La méthodologie pour identifier les besoins des utilisateurs en termes de la QoS consiste à remplir le Tableau 2-10 qui représente une matrice complète à partir de laquelle nous pourrons obtenir des résultats pertinents.

Le critère de Qualité			**Vitesse**	**Précision**	**Disponibilité**	**Fiabilité**	**Sécurité**	**Simplicité**	**Flexibilité**
La fonction du Service		**Réf.**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Les ventes		**1**							
Le Management du Service	**Disponibilité**	**2**							
	Modifications	**3**							
	Le support	**4**							
	Réparation	**5**							
	L'arrêt	**6**							
Les paramètres techniques d'un appel	**L'établissement de la connexion**	**7**							
	Information sur le transfert	**8**							
	L'émission	**9**							
La facturation		**10**							
Le management du service par le client		**11**							

Tableau 2-10 Matrice pour faciliter la prise en compte des exigences des utilisateurs

La deuxième partie du document nous donne des indications sur les paramètres à mesurer en fonction des exigences des utilisateurs. La difficulté réside dans le fait que ces paramètres ne tiennent pas compte du support technique utilisé par le fournisseur. Si nous prenons l'exemple de la téléphonie, nous utilisons actuellement comme support l'ATM, le frame relay, l'IP etc et donc pour chacun de ces supports, il est impératif de trouver les paramètres «communs » à tester qui dérivent des exigences des utilisateurs. Ces paramètres représentent une base pour la définition des Service Level Agreement qui sera détaillée dans la troisième partie du document.

Résumé EG 202 009

Le document présente une méthodologie pour identifier les paramètres significatifs pour l'utilisateur final afin de contrôler la QoS pour des services télécoms ou pour produire des SLAs (Service Level Agreement) entre le client et le fournisseur de services. Il donne des conseils sur la manière de traduire les exigences des utilisateurs en termes de paramètres de QoS, pour déterminer une valeur préférentielle pour chacun de ces paramètres. Nous trouvons aussi des paramètres spécifiques aux différents services. Pour la première fois, nous sommes face à un document qui prend en compte le client et ses besoins en termes de QoS, mais comme ses auteurs eux-mêmes le font remarquer cette première version ne représente juste qu'un début.

2.5.6 Les Normes IETF

L'IETF, Internet Engineering Task Force, représente le groupe qui s'occupe de la standardisation au sein de la Société Internet [IETF00]. C'est une grande communauté internationale de concepteurs, d'opérateurs, de fournisseurs et de chercheurs concernés par l'évolution de l'architecture d'Internet et toutes opérations qui s'y rapportent. Le travail technique concret de l'IETF s'effectue dans ses groupes de travail, organisés par matière et secteurs (par exemple, routage, transport, sécurité etc...).

Les principaux modèles concernant la QoS proposés par l'IETF ont été traités dans la Section 2.3.4 « Qualité des Service dans le monde IP ».

2.6 Résumé Du Chapitre

Les conditions pour mesurer la qualité de service sont définies par le marché (c'est à dire les utilisateurs et les industriels). Prises ensemble ces demandes vont créer un besoin de définition, mais aussi un moyen rentable pour l'évaluation de la qualité, une harmonisation et une certification de la qualité des produits, des processus et des services.

Le Tableau 2-11 résume le travail effectué sur les différentes normes dans ce chapitre.

	ISO 9000	**ISO 9126**	**I.350**	**X.641/ X.642**	**X.900**	**EG 202 009**
Orientation du cadre						
Focalisation sur l'utilisateur final	++	+	++	-	++	++
Evaluation versus gestion	++	+	++	+	-	+
Orientation vers l'application	+	+	--	+	++	++
Méthode d'évaluation	0	0	-	0	0	+
Les caractéristiques de la Qualité						
La perception de l'utilisateur final	-	+	++	--	++	++
Focalisation sur quelques caractéristiques	--	--	-	--	-	++
Les propriétés de la méthode d'évaluation						
L'appréhension	0	0	0	0	0	+
L'autonomie	0	0	0	0	0	++
L'automatisation	0	0	0	0	0	0
La facilité d'utilisation	0	0	--	0	0	++
L'évaluation des résultats						
La comparabilité	+	0	+	0	0	+
La reproductibilité	+	0	+	0	0	+

Tableau 2-11 Les normes internationales examinées concernant QoS.

++: très bon, +: bon, 0: non pertinent ; -: pauvre ; --: très pauvre

Dans notre comparaison des cadres d'évaluation de la qualité de service, nous avons pris en compte le point de vue d'un utilisateur de services multimédia. Les utilisateurs doivent participer à l'évaluation de la qualité de service. Afin d'éviter de surcharger un utilisateur, nous avons en plus souhaité une limitation du nombre des critères d'évaluation de la

qualité de service en éliminant les éléments concernant la disposition du service, l'accès au service et enfin la qualité du réseau.

Les normes examinées, à l'exception de la norme EG 202 009 ne fournissent pas d'éléments de la qualité perçue par l'utilisateur final. On ne trouve pas dans ces documents (sauf dans I.350) de méthodes pour l'évaluation de la qualité de service et des rapports sur le sujet avec les résultats obtenus (comparables et reproductibles si possible).

Le travail de l'IETF ne fournit pas une approche conforme à notre objectif; ISO 9000 est plus orienté vers la production tandis que ISO 9126 est un standard de qualité du logiciel, bien qu'il soit potentiellement très utile. La même remarque s'applique à I.350 tandis que les séries X des recommandations sont des cadres puissants qui doivent être adaptés, éclairés et raffinés pour convenir à nos conditions. Seul le document de l'ETSI prend en compte les besoins des utilisateurs, mais pour le moment les notions sont encore trop généralistes.

Les normes décrites dans cette section ne remplissent pas les conditions requises pour servir à des utilisateurs finaux. Elles ne sont pas conçues pour résoudre les problèmes identifiés, elles ne concernent pas les préoccupations spécifiques d'un utilisateur ou elles sont trop compliquées à employer.

Dans ce qui suit, nous allons utiliser la terminologie présentée. Les notions concernant la qualité de service issues des normes internationales représenteront un point de départ pour notre méthodologie d'évaluation de QoS. Le cadre de cette évaluation est le sujet de notre prochain chapitre.

3 Un cadre d'évaluation de la Qualité de Service

Le chapitre 3 présente le cadre d'évaluation de la qualité du service développé pendant cette thèse. Il se focalise sur l'utilisateur final. Ce nouveau cadre se compose d'un procédé simple d'évaluation de la Qualité de Service, une introduction aux Agents Intelligents et un modèle d'Agent Intelligent qui supporte une évaluation de la QoS.

3.1 Le Modèle d'un Mécanisme pour l'Evaluation de la qualité du service

L'information fournie à un client est généralement différente de celle employée par un fournisseur de service déployant un service. Elle est moins détaillée, car les clients sont intéressés par l'information concernant leur propre service et non pas par tous les détails techniques du déploiement de service. Afin de permettre l'évaluation de la qualité de service par les utilisateurs finaux, il faut réduire le nombre de propriétés d'une part et mettre en place de méthodes de mesure appropriées d'autre part. Dans la section 2.2.3, « La liste des souhaits d'un utilisateur pour l'évaluation de la Qualité de service », aucune description détaillée n'a été faite pour préciser quel type de procédé d'évaluation de la qualité de service est nécessaire et à quoi les résultats d'évaluation réalisés devraient ressembler. Avant de montrer comment notre procédé (qui sera développé dans ce chapitre) remplit ces souhaits nous devons d'abord identifier les acteurs et leurs rôles dans notre structure d'évaluation de la qualité de service.

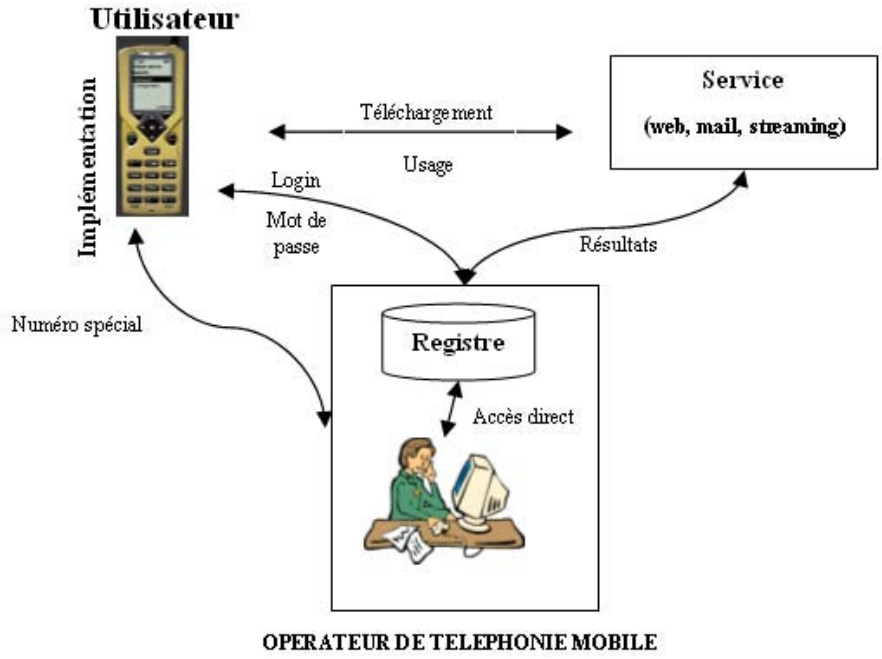


Figure 3-1 Modèle d'un Mécanisme pour l'évaluation de la qualité du service sur les mobiles.

Notre modèle (Figure 3-1) pour les parties suivantes de cette thèse contient *un Utilisateur, un Registre, une Implémentation* et naturellement *une Application/Service* pour tester. L'utilisateur final veut ou doit faire une évaluation de la qualité de services proposés sur son mobile. Comme il n'est pas un expert, il découvre que les outils d'évaluation de la QoS sur le marché ne l'aideront pas (voir la section 1.1.3). L'utilisateur final sera dérouté par la complexité des normes, par les caractéristiques techniques pour la disposition de service et par le nombre de caractéristiques concernant la qualité. En conséquence, la seule façon qu'il a de tester la QoS est de voir si les applications qui lui sont proposées fonctionnent. Dans le meilleur de cas, la réponse est positive, mais comment et dans quelles conditions? L'utilisateur essayera de trouver de l'aide ailleurs. Il s'adressera à des professionnels du domaine de la qualité de service ou à l'aide d'Internet. Dans le meilleur des cas, il se retrouvera dans des bases de données, mais son problème restera toujours non résolu. Les services clientèle actuels opérateurs sont incapables de proposer ce genre de service. C'est une des raisons pour lesquelles, nous essayons de mettre à la disposition des utilisateurs finaux un nouveau modèle d'évaluation de la qualité de service.

3.1.1 L'Utilisateur final

Dans le modèle pour l'évaluation de la qualité de service les propriétés et les tâches suivantes caractérisent *l'utilisateur final* :

- il possède un téléphone mobile Java SimToolkit (voir la section 2.3.5) qui accepte des applications J2ME [Réf. J2ME] ;

- il est l'utilisateur d'un service, il veut devenir un utilisateur de service ou il agit au nom d'autres utilisateurs de service ;

- les services auxquels il a accès se trouvent dans le contrat de service (SLA – Service Level Agreement) qu'il a signé avec son opérateur de téléphonie mobile ;

- il souhaite faire une évaluation de la qualité de service et il cherche une solution appropriée ;

- il possède un compte d'accès à la base de donnés de l'opérateur (voir section suivante *le Registre*). L'accès est sécurisé par un nom d'utilisateur (par exemple user@nomopérateur.fr), par un mot de passe et se fait par l'intermède du site Internet de l'opérateur ;

- il trouve en téléchargement l'application concernant l'évaluation de la qualité de service ;

- il peut appeler à n'importe quel moment le service clientèle pour avoir les renseignements nécessaires concernant l'installation et le mode d'emploi de l'application ;

- il fait l'évaluation de la qualité de service (voir la section 3.2.5) sur les services qu'il souhaite à l'aide de l'application chargée sur son mobile ;

- il reçoit les résultats de l'évaluation de la qualité de service sur son mobile (voir la section 5.7) [1];

- il laisse des avis et des recommandations pour d'autres utilisateurs dans le Registre.

3.1.2 L'opérateur de téléphonie mobile

Dans le modèle pour l'évaluation de la Qualité de service les propriétés et les tâches suivantes caractérisent *l'opérateur de téléphonie mobile* :

- il possède un réseau d'accès mobile ;

- il possède une base de données des clients (le Registre) qui peut être accessible en toute sécurité par l'intermédiaire du site web ;

- il a un service clientèle disponible par téléphone (numéro spécial) ou sur la rubrique FAQ (Frequently Asked Questions) sur le site web ;

- il négocie des contrats de service (SLA – Service Level Agreement) avec ses clients ;

- il est à l'écoute de ses clients et il propose les supports nécessaires pour l'évaluation de la qualité de service ;

- il a accès à différentes méthodes certifiées d'évaluation de la qualité de service ;

- il est responsable de la certification des méthodes d'évaluation de la qualité de service ;

- il possède une bibliothèque Java de modules standard de certification pour les méthodes d'évaluation de la qualité de service.

[1] Des résultats plus détaillés et comparables avec des données contractuelles se trouvent dans le Registre. Pour des raisons de taille et d'ergonomie du téléphone mobile ces résultats ne peuvent pas être consultés que par l'intermède du site web de l'opérateur.

- il lance le développement de nouvelles des méthodes d'évaluation de la qualité de service basées sur des demandes d'évaluation de la part des utilisateurs ;

- il reçoit le retour des résultats d'évaluation de la Qualité de service des utilisateurs, il les rassemble et il crée une base de données ;

- il peut exécuter des essais indépendants avec les méthodes d'évaluation de la qualité de service disponibles ;

- il lance le perfectionnement des méthodes d'évaluation de la qualité en s'appuyant sur le retour des utilisateurs ;

- il peut négocier avec les fournisseurs de service au sujet de l'évaluation de la qualité de service (par exemple : les droits d'accès, les pré-requis, la sécurité et le paiement) ;

- il fournit et il publie les informations, les résultats d'évaluation de la qualité de service aux utilisateurs et aux fournisseurs de service.

3.1.3 Registre

Dans le modèle pour l'évaluation de la qualité de service les propriétés et les tâches suivantes caractérisent *le Registre* :

- il représente une base de données, propriété de l'opérateur de téléphonie mobile ;

- il est accessible par un login et un mot de passe dans le cas d'un client ou directement par le service clientèle. L'accès est fait soit par le mobile, soit par le site Internet de l'opérateur ;

- il contient des informations sécurisées, non modifiables et qui sont uniquement en consultation ;

- chaque accès dans la base est enregistré.

Après avoir introduit les acteurs et leur rôle dans notre cadre de travail, nous allons présenter dans la section suivante un modèle simple de l'évaluation de la QoS. Ce modèle

sera appliqué par la suite à chaque service sur le mobile que nous allons tester (accès web, mail et streaming vidéo).

3.2 Un procédé en 5 pas pour l'évaluation de la Qualité de Service

L'évaluation de la qualité est souvent considérée comme une procédure trop complexe pour contrôler tous les sujets et les attributs potentiels. Cette approche n'est pas très efficace parce qu'elle n'aide pas à identifier des critères principaux qui aident à prendre les décisions pertinentes. Notre but est de faciliter la tâche des utilisateurs. La solution proposée est la suivante : dans un premier temps nous devons identifier le service à évaluer (par exemple, le service de courrier électronique) et les composants de base de ce service. Ensuite, nous devons rechercher les caractéristiques de ces composants, les quantifier et les mesurer. Pour avoir des résultats concrets, on a simulé le processus d'évaluation.

La méthode d'évaluation de la qualité de service décrite dans l'étape suivante est basée sur des méthodes établies dans des logiciels d'évaluation de la qualité du début des années 90 sur la base de IS09126 [Réf ISO 9126]. Pour nous rapprocher du client final, nous nous sommes inspirés aussi de la méthode d'identification des exigences des clients qui est apparue dans la norme ISO 9000, version année 2000 [Réf. ISO9000].

La Figure 3-2 donne une image globale de la procédure que l'on propose. Elle ressemble à une procédure dans un réseau de Pétri, où les cercles (les attributs dans la terminologie du réseau Pétri) représentent les entrées et les sorties d'une étape de la procédure représentée par une boîte (une transition dans la terminologie du réseau Pétri).

Différentes caractéristiques et recommandations peuvent être employées en tant que point de départ pour ce procédé d'évaluation de la qualité service :

- Une description du produit logiciel SPD (Software Product Description), (par exemple l'installation et le fonctionnement de notre application) ;

- Un document avec des spécifications technique TSD (Technical Specification Document) comme [ISO 11172] ;

- Une partie de l'accord sur le niveau de service SLA (Service Level Agreement) entre l'opérateur et le client ;

- Un standard de la qualité du logiciel SQS (Software Quality Standard) comme ISO 9126 [Réf. ISO9126].

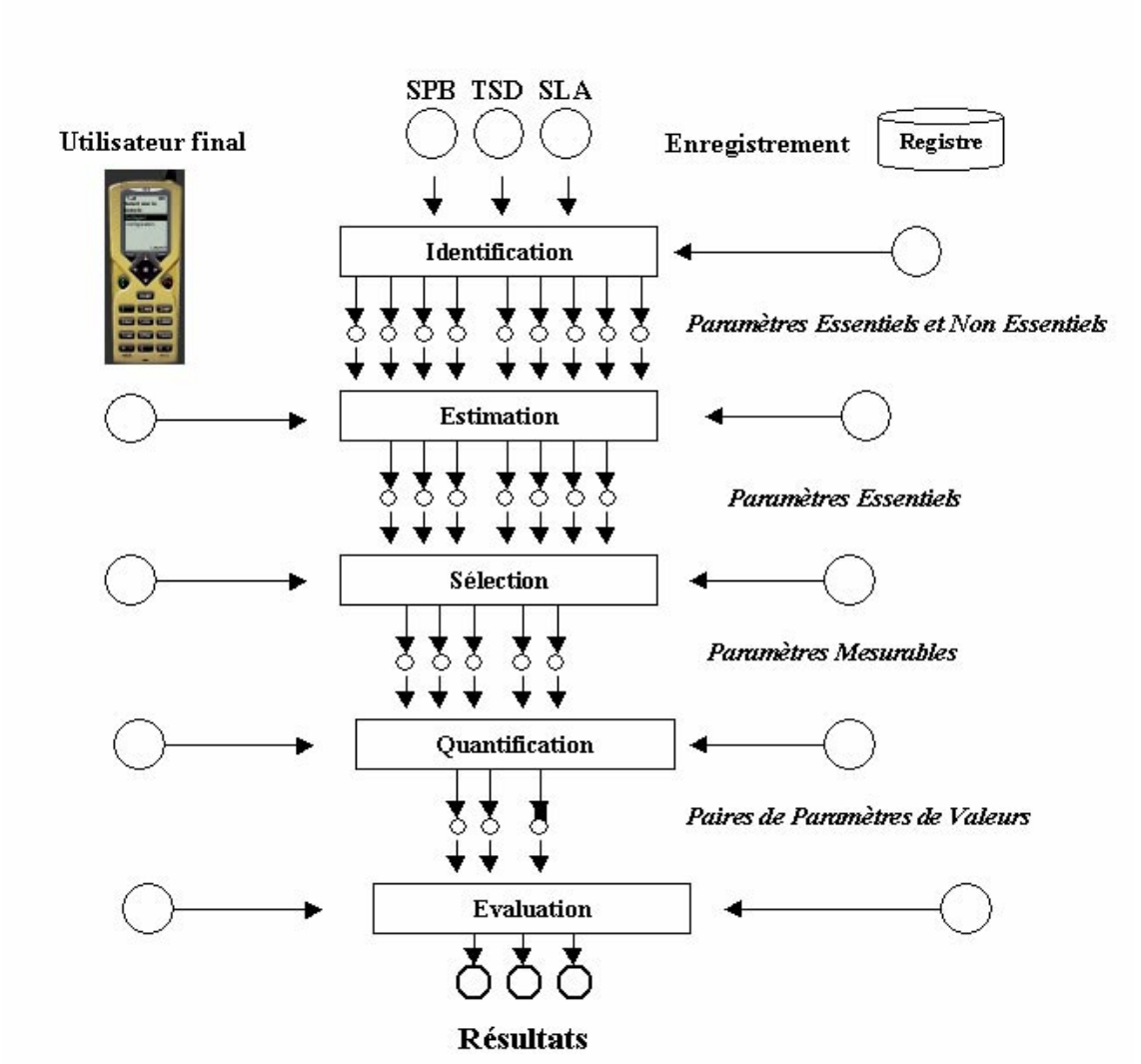


Figure 3-2 Procédé d'évaluation de la qualité service.

Les sorties finales de cette procédure représentent les résultats d'évaluation. Dans la suite, nous présentons chaque étape de cette procédure d'évaluation.

3.2.1 Etape 1 : Identification de tous les paramètres de la qualité

L'évaluation de la qualité de service devrait être effectuée par chaque utilisateur. Des outils appropriés d'évaluation doivent être mis en place pour les soutenir. Le but est d'identifier les paramètres essentiels et non essentiels de la qualité. Le problème dans cette étape est le nombre et la complexité de paramètres présents.

Dans la section 2.2, nous avons remarqué qu'un utilisateur ne peut pas et ne voudra pas avoir une compréhension détaillée de la technologie impliquée dans la disposition de service. C'est là qu'intervient l'expérience de la personne qui sera chargé de développer la procédure d'évaluation de la qualité de service. Celle-ci possède les compétences nécessaires pour comprendre le fonctionnement technique de cette procédure. Dans notre cas, l'outil d'évaluation qui sera développé est basé sur l'expérience acquise au bout de 3 ans de recherche dans le domaine de la qualité de service. Dans le cas d'un opérateur de téléphonie mobile, la démarche sera faite en fonction de l'évolution des services proposés et à la demande des ses propres clients adaptés à leurs besoins et attentes.

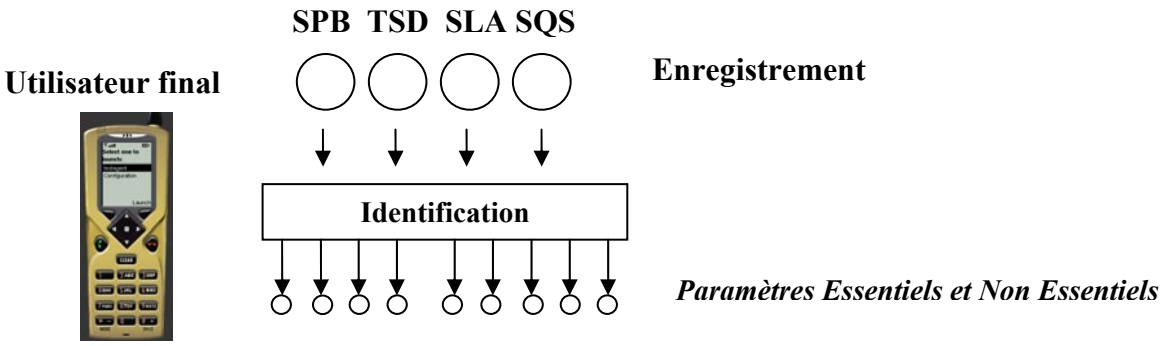


Figure 3-3 Etape 1 : Identification de tous les paramètres de la qualité

Dans cette étape, nous devons identifier tous les paramètres Essentiels et non obtenus en étudiant les spécifications et les recommandations impliquées. L'identification est faite à l'aide des entrées énumérées dans la Figure 3-3. Comme nous ne pouvons pas traiter de réels contrats de services (SLA), nous sommes obligés de nous appuyer sur les normes actuelles et sur notre propre expérience. Cette première étape nous permettra, en tant que

développeur de l'application, de nous situer par rapport au service que nous souhaitons évaluer (accès web, mail, etc.).

Dans le cas de l'opérateur, une liste de paramètres essentiels ou non essentiels de la qualité pour chaque service, doit se trouver dans le Registre. L'utilisateur final n'est pas forcement concerné par cette liste. Par contre, un document contenant des indications concernant le téléchargement, la mise en place et l'utilisation de l'application pour l'évaluation de la QoS est nécessaire, ainsi que son contrat de service avec l'opérateur. L'évaluation de la qualité de service peut être faite par un utilisateur ou à l'initiative de l'opérateur qui souhaite tester certains de services proposés.

3.2.2 Etape 2 : Le classement

Quand nous parlons des applications multimédia, parmi les paramètres de la qualité de service mentionnés, il faut faire la différence entre les paramètres essentiels et les non essentiels. Une sélection ou une réduction pour celles qui sont essentiels et mesurables doit avoir lieu. Le classement est assez subjectif, il dépend des conditions de l'environnement de l'utilisateur. D'une part, si l'utilisateur final se trouve dans un endroit sans avoir un accès direct au site Internet de l'opérateur, les informations qui lui seront envoyées sont assez limitées à cause des capacités du réseau et de la taille de l'écran de son téléphone mobile. D'autre part, l'utilisateur final n'est pas un expert en matière de QoS. Par conséquent, il est important dans cette étape de fournir à l'utilisateur une vraie possibilité d'indiquer son estimation sans le surcharger avec les détails techniques. Il devrait être possible que par exemple, dans un processus d'évaluation d'une vidéo en streaming, l'utilisateur puisse indiquer la qualité visuelle comme telle, sans avoir besoin d'indiquer des paramètres comme le taux de trames, la chrominance, la luminance ou le débit binaire. Une comparaison par la suite avec des vrais paramètres peut éclaircir et guider l'utilisateur pour une prochaine évaluation. Une bonne interface sera cruciale pour cela et la participation de l'utilisateur est obligatoire dans cette étape.

Dans cette deuxième étape, les conditions et l'environnement spécifiques d'un utilisateur sont pris en compte pour trouver les caractéristiques essentielles de la qualité. L'expertise et l'expérience disponibles à chaque enregistrement soutiennent ceci. Elles sont fusionnées pour déterminer le résultat de cette étape de classement.

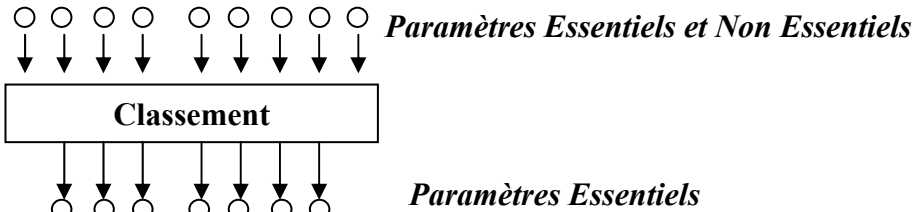


Figure 3-4 Etape 2: Le classement

Dans le Figure 3-4 les contraintes d'un utilisateur, les connaissances et l'expérience de l'opérateur sont décrites en tant qu'entrée additionnelle pour cette étape. Cette évaluation commune fait la liste des caractéristiques de la qualité exclusive et adaptée aux besoins d'un utilisateur. Elle peut inclure aussi des expériences, des évaluations et des résultats d'autres utilisateurs.

3.2.3 Etape 3 : Choix des Mesures

Dans cette troisième étape, un choix doit avoir lieu afin d'identifier les caractéristiques essentielles de la qualité qui peuvent être mesurées d'une manière automatique. Les caractéristiques subjectives de la qualité ne peuvent pas être déterminées automatiquement et directement. Il faut les traduire et les lier aux paramètres objectifs de la QoS d'un service pour qu'elles deviennent mesurables. L'entité qui effectue la demande, un utilisateur par exemple, doit faire son choix en s'appuyant sur ses propres besoins.

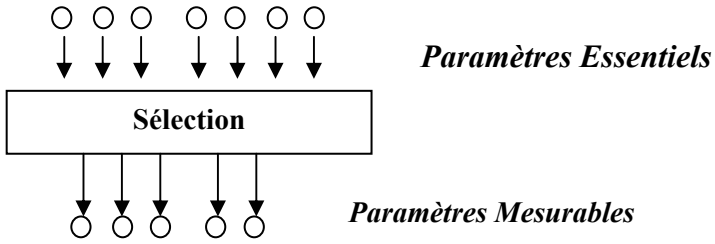


Figure 3-5 Etape 3 : Choix des Mesures

Le problème dans cette étape est de trouver un accord entre l'utilisateur et l'opérateur au sujet des caractéristiques de la qualité qui peuvent être subjectives, objectives, faciles ou difficiles à mesurer. Le contrat de service (SLA) qui a été négocié entre l'opérateur et ses clients a un rôle déterminant dans cette étape. Ce contrat représente une base utile pour le choix de paramètres de la QoS à mesurer. Même si les utilisateurs sont intéressés par la qualité de différents services qui leur sont proposés, leur choix est réduit par rapport au contrat qu'ils ont signé avec leurs opérateurs.

Le développeur du logiciel d'évaluation de la qualité de service a un apport important dans le choix de paramètres. Celui-ci donnera un avis technique sur la faisabilité et la mise en œuvre de ces paramètres de mesures.

3.2.4 Etape 4 : Quantification des Paramètres mesurables

La QoS d'un service donné peut être exprimé par un ensemble de paires paramètres valeurs. Dans cette quatrième étape, une paire de paramètre-valeur pour chaque caractéristique de la qualité mesurée doit être définie. L'utilisateur apporte ses contraintes, tandis que l'opérateur y contribue avec des résultats et des recommandations d'anciennes évaluations. Dans cette étape, nous pouvons constaté l'importance du SLA (Service Level Agreement) pour déterminer ces valeurs clefs.

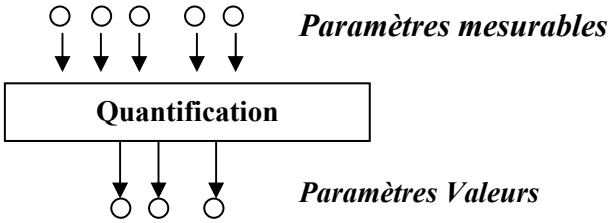


Figure 3-6 Etape 4: Quantification dans des paires Paramètres Valeurs

L'opérateur doit s'appuyer sur l'expérience et l'expertise à chaque enregistrement pour trouver un ensemble de paires de paramètre-valeur adapté aux besoins du client. Le contrat de service doit être établi en conséquence par rapport aux besoins du client. Cette étape détermine des valeurs numériques qui serviront dans la prochaine étape comme valeurs de seuils pour l'évaluation de la qualité de service.

3.2.5 Etape 5 : L'évaluation de la qualité de service

Dans l'étape 4, nous avons défini les caractéristiques essentielles mesurables de la qualité ainsi que les conditions limites et les paires de paramètre-valeur pour que l'évaluation soit exécutée dans cette étape. Mesurer une caractéristique objective de la qualité signifie appliquer une certaine méthode de mesure et comparer les mesures obtenues à des valeurs requises (seuils). Les résultats obtenus sont inscrits dans un Registre qui est accessible par l'utilisateur final.

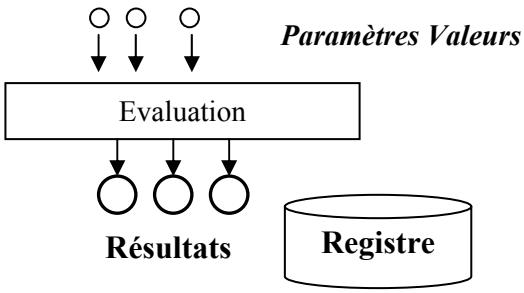


Figure 3-7 Etape 5 : Evaluation de la qualité de service

Si une méthode d'évaluation de la Qualité de service est déjà disponible ou connue à un enregistrement (accès par le site Internet de l'opérateur), l'évaluation peut être faite par l'utilisateur immédiatement. Autrement, l'opérateur doit lancer son implémentation ou trouver une méthode d'évaluation appropriée.

Il est important que les résultats obtenus après l'évaluation soient compréhensibles par les clients. Ceux-ci ne comprennent pas toujours les données numériques, donc il sera intéressant de présenter ces résultats sous forme d'une note d'appréciation (MOS – Mean Opinion Score) [Réf ITU P800]. Cette note peut se traduire sous forme d'une échelle de *Mauvais à Excellent* (voir Tableau 3-1).

5	**EXCELLENT**
4	**BON**
3	**JUSTE**
2	**PAUVRE**
1	**MAUVAIS**

Tableau 3-1 Note d'appréciation

Cette échelle sera établie au cours de la dernière étape de notre évaluation par la comparaison de la valeur obtenue à la valeur du seuil proposé dans le SLA.

Des méthodes d'évaluation et les valeurs de paramètre respectives exigées (spécifications de la QoS) d'un service sous test peuvent être stockées dans la base de données de l'opérateur (le Registre). Elles devraient être accessibles par n'importe quel client, même à distance.

Dans le Tableau 3-2 la procédure d'évaluation de la qualité de service est récapitulée.

Etape	Action	Signification
1	*Identification*	Toutes les caractéristiques de la qualité (c.-à-d. Essentielles et des Non –Essentielles) doivent être identifiées.
2	*Classement*	Dans la liste de toutes les caractéristiques de la qualité trouvées, un classement est effectué entre les caractéristiques essentielles et non essentielles
3	*Choix*	On détermine quelles sont les caractéristiques de la qualité qu'il faudra mesurer.
4	*Quantification*	Pour chaque caractéristique choisie il faut trouver les paramètres- valeurs (seuils) qui existent dans les SLA.
5	*Evaluation*	L'évaluation doit être faite un prenant en considération les paires de paramètres- valeur indiquées.

Tableau 3-2 Une procédure en 5 étapes d'évaluation de la qualité de service

Maintenant que nous avons présenté le procédé d'évaluation de la qualité de service, nous devons parler de sa mise en place. Celle-ci est conçue pour soutenir les utilisateurs des services multimédia.

3.3 Une Architecture d'Evaluation de la Qualité de Service

L'évaluation de la QoS doit être automatisée pour être réussie et fournir des résultats utilisables. Nous proposons donc une architecture appropriée, faisant participer des utilisateurs, des méthodes d'évaluation ainsi que des applications et des services en test. L'utilisation des Agents Intelligents permet d'automatiser l'évaluation de la qualité de services pour des applications et des services.

3.3.1 Qu'est ce qu'un Agent ?

Le domaine des Agents est assez récent. Les premières tentatives remontent à quelques années et le terme d'Agent logiciel est encore mal défini. Si nous nous référons à la définition du dictionnaire : agent du latin : « agens », celui qui agit.

« Un Agent est une personne chargée des affaires et des intérêts d'un individu, d'un groupe ou d'un pays, pour le compte desquels il agit».

Le terme « agir » est défini par le petit Larousse comme suit : agir du latin « agere » faire, faire quelque chose, s'occuper, produire un effet.

De ces définitions, nous retirerons deux aspects fondamentaux : un Agent accomplit quelque chose et un Agent agit à la demande de quelqu'un (Agent ou utilisateur).

Concernant les agents logiciels nous nous sommes arrêtés sur la définition donnée par Caglayan et Harrisson [Réf Alper97] : « *Agent logiciel : entité informatique qui réalise de manière autonome des tâches pour un utilisateur. »*

Dans notre cas, l'agent logiciel (plus exactement une application Agent) devra faire quelque chose pour une personne ou une application. Plus exactement une application Agent sera orientée tâche, c'est-à-dire qu'elle déploiera une activité, (suite de fonctionnalités offertes par son environnement), dans le but de faire quelque chose et sera caractérisée par un certain degré d'autonomie, d'interactivité et de réactivité.

Pour mieux comprendre la notion d'Agents Intelligents il faut définir la notion d'intelligence. Concernant l'intelligence des agents de nombreux auteurs ont donné leurs visions des choses mais aucune de ces visions n'est reconnue universellement (un auteur = une définition). Une définition assez intéressante a été donné par J.F. David [Réf David97] « *L'intelligence d'un système plongé dans un environnement complexe peut être définie comme son temps de réponse entre la détection du besoin exprimé (ou du désir latent) de son " client " et la réponse adéquate à ce besoin.* ». Le même auteur précise que en fait « *qu'il y a deux faces à l'intelligence d'un agent : Celle tournée vers l'utilisateur, afin de comprendre et d'adapter sa mission, et celle tournée vers le monde extérieur, afin de l'accomplir.* ». Nous avons suivi cette optique pour développer notre agent pour l'évaluation de la qualité de service pour des services multimédia. Ces services sont développés sur des systèmes à plein temps (systèmes informatiques, système de gestion etc.) qui peuvent être classifiées suivant leur "niveau d'intelligence" [Réf David97] :

- niveau 0, le système n'a aucune réaction aux événements ;
- niveau 1, le système est capable de réactions sélectives, comme le thermomètre qui monte et descend en fonction de la température (système régulé) ;

- niveau 2, le système est doté de mémoire pour accumuler son expérience et d'organes de décisions capables de puiser dans cette mémoire, afin d'avoir des réactions sélectives à son environnement ;
- niveau 3, le système peut de plus s'auto-inventer des objectifs et nous sommes alors au niveau de l'individu ou de l'entreprise. A ce moment, la notion de « but » est très importante et peut être intégrée dans le système ou être dynamiquement adaptée en fonction de l'environnement.

Après avoir déterminé toutes ces notions, nous pouvons dire que les agents intelligents peuvent être définis comme des entités logicielles dotées de suffisamment d'autonomie décisionnelle pour effectuer des tâches spécifiques avec peu ou pas de surveillance humaine. Un agent agit dans un environnement en constante évolution tout en représentant les intérêts d'un propriétaire particulier.

3.3.2 Caractéristiques des Agents

L'intelligence d'un Agent peut être vue aussi comme un agrégat de caractéristiques [Réf. H.NWA] : la capacité d'apprendre, la capacité sociale et un haut degré d'autonomie. L'ensemble des caractéristiques des Agents avec leur définition sont présentées dans la tableau suivant (Tableau 3-3):

Propriété	***Signification***
Autonome	contrôle ses propres actions
Téléonomique	ses actions ne sont pas de simples réactions à l'environnement
Temporellement persistant	travaille sans arrêt
Communiquant	communique avec d'autres agents, dont certains du monde réel
Apprenant	adapte son comportement en fonction de ses expériences
Mobile	capable de se transporter d'une machine à l'autre
Flexible	actions non totalement programmées

Tableau 3-3 Propriétés des Agents

3.3.3 Types d'Agents

A part les Agents Intelligents, d'autres types d'Agents peuvent être définis par rapport à ces propriétés. Ceux-ci peuvent être classées conformément à la [Réf OMG-AWG] en :

3.3.3.1 Les Agents mobiles

Ces agents se déplacent d'une machine à l'autre en transférant leur code et leurs données sur le nouveau site et en continuant son exécution sur ce site là. L'intérêt de ces Agents est la recherche des données sur des sites éloignés. Un exemple de ce type est d'Agent GRASSHOPPER [Réf Grass] qui sert à la gestion d'un réseau intelligent (Intelligent Network).

3.3.3.2 Les Agents d'information

Ce type d'Agent peut être divisé en 5 sous types :

- *« Interest Matching Agents »* qui sont des agents très utilisés sur les sites commerciaux pour mettre en valeur des produits qui ressemblent au produit venant d'être acheté par un utilisateur ;

- Les Agents de recherche d'information qui sont des Agents fiables pour retrouver des informations utiles ;

- Les Agents de filtrage d'information qui ont comme rôle de trier les informations arrivant à l'utilisateur ;

- Les Agents de suivi de l'information qui permettent de consulter des sources de données diverses et de faire un suivi permanent des changements apportés ;

- Les Agents de médiation de sources de données qui peuvent être employés comme des médiateurs entre toutes ses sources de données, fournissant les mécanismes leur permettant d'inter-opérer.

3.3.3.3 Les Agents d'interface utilisateur

Un Agent d'interface est un programme qui assiste l'utilisateur dans la mise en œuvre d'un système ou d'un logiciel (Voir les Agents office de Microsoft). Dans le futur, ces Agents pourront être étendus au monde du WEB, pour assister l'utilisateur.

3.3.3.4 Les Agents de l'analyse de l'offre

Ces agents sont de plus en plus présents sur Internet. Ils sont utilisés pour optimiser la recherche d'informations, pour animer un site web, pour dialoguer avec les internautes, les aider à faire des choix…Ils se substituent aux interfaces actuelles existant entre l'homme et l'ordinateur en « humanisant » la relation entre les internautes, entre les consommateurs et les vendeurs. Quelques exemple **BargainFinder** de Andersen Consulting ; **Jango** intégré dans Excite et **Firefly** intégré dans Yahoo.

Pourquoi utiliser des Agents Intelligents ? Voici quelques réponses :

- tout d'abord l'évolution des services qui deviennent de plus en plus riches et difficiles à maîtriser par les utilisateurs ;
- l'accroissement des sources d'informations ;
- l'augmentation de la puissance des machines actuelles permet de plus en plus d'intelligence « locale » à la portée de l'utilisateur [16] ;
- les problèmes deviennent de plus en plus complexes dans le cas des applications multimédia. (comme nous l'avons déjà souligné nous sommes dans des environnements distribués où les problèmes d'interconnexion et interopérabilités sont de plus en plus complexes et un seul programme informatique ne suffit plus à gérer la situation) ;

[16] Dans le cas d'un réseau GSM, la bande passante reste occupée pendant la communication. Les opérateurs GSM ont tendance à augmenter les bandes passantes de leurs réseaux en raison d'une grosse saturation pendant les heures de pointe.

- les bandes passantes des réseaux augmentent, mais leur utilisation est limitée dans le temps ;

Pour mieux intégrer les Agents Intelligents dans notre travail, il est important de connaître leur mode de fonctionnement. C'est l'objet du paragraphe suivant.

3.3.4 Fonctionnement de l'Agent Intelligent

Dans la [Réf Bonnet], nous trouvons des explications concernant le fonctionnement d'un Agent. Pour mettre en évidence les principales caractéristiques d'un AI (Agent Intelligent): la capacité sociale, la capacité d'apprendre et le degré d'autonomie, il faut intégrer au programme agent les modules suivants :

· *Un module permettant de mémoriser le savoir de l'agent qui sera le plus souvent fait par la réalisation d'un système de base de données (SGBD). Ce module sera nommé « base de connaissance ».*

· *Un module permettant de connaître l'ontologie de la « base de connaissance ». Il sera appelé « Le contrôleur d'accès » qui peut reposer sur un SGBD, dit « intelligent. »*

· *Un module permettant à l'agent d'avoir une vision de son environnement, « l'interface de l'agent. »*

· *Un module permettant la génération de nouvelles connaissances, il sera nommé « moteur d'apprentissage.»*

· *Un module permettant le raisonnement de l'agent, il sera nommé « moteur de raisonnement. »*

Ces modules devront être agencés de façon à ce que le flux d'information qu'ils génèrent, permette de caractériser le concept d'intelligence. Un des modèles le plus souvent rencontré est représenté par le schéma suivant (Figure 3-8):

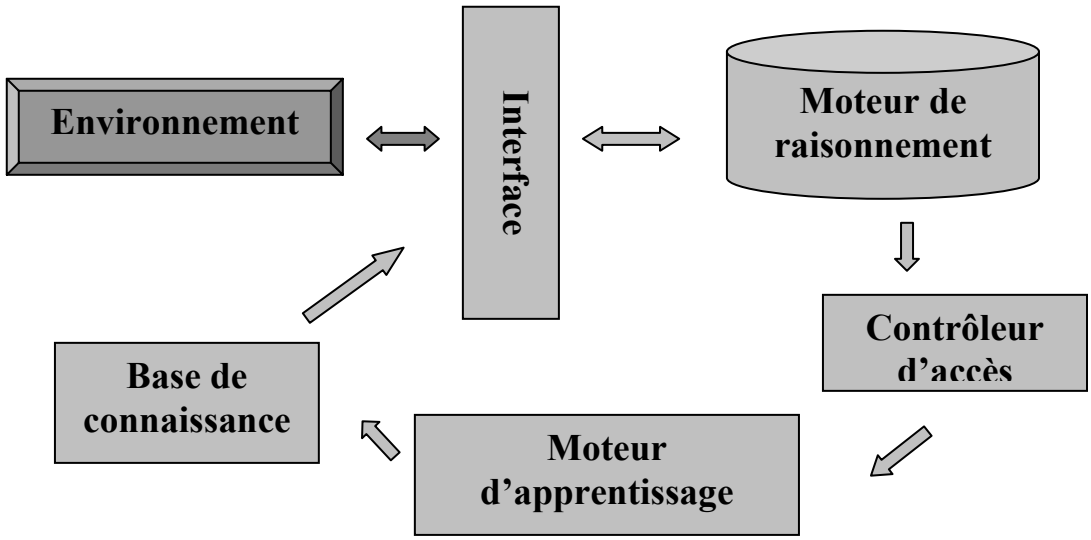


Figure 3-8 Mécanisme de l'intelligence chez un agent

Nous pouvons dès lors, faire la distinction entre quatre groupes [Réf CAG97] :

- Le contenu représente les données employées par les mécanismes dans le raisonnement de l'apprentissage (représenté sur le schéma par la base de connaissance) ;
- Le mécanismes de l'agent contenant les moteurs d'apprentissage et de raisonement ;
- L'accès qui permet de percevoir le contenu des connaissances de l'agent ;
- L'interface permettant une perception de l'environnement de l'agent.

Les trois premiers groupes reposent techniquement, dans la plupart des cas, sur un moteur d'inférence. Ce dernier contient : les domaines et les relations entre les « objets » utilisés (« l'accès »), les faits (« Le contenu »), ainsi que des règles (« les mécanismes »).

Quand à l'interface, elle repose le plus souvent sur une application de type « middleware[17] » permettant d'agir et de prendre des informations sur l'environnement de l'Agent.

Les Agents solos ont très peu d'interaction avec d'autres Agents et sont restreints par leurs propres limites. Les systèmes Multi-Agents (MAS : Multi Agent System) peuvent profiter des divers rôles de chacun des Agents du système.

3.3.5 Une structure d'un système Multi-Agents

Dans le cas de notre recherche, nous sommes dans la situation d'un système Multi-Agents qui peut être schématisé de la manière suivante :

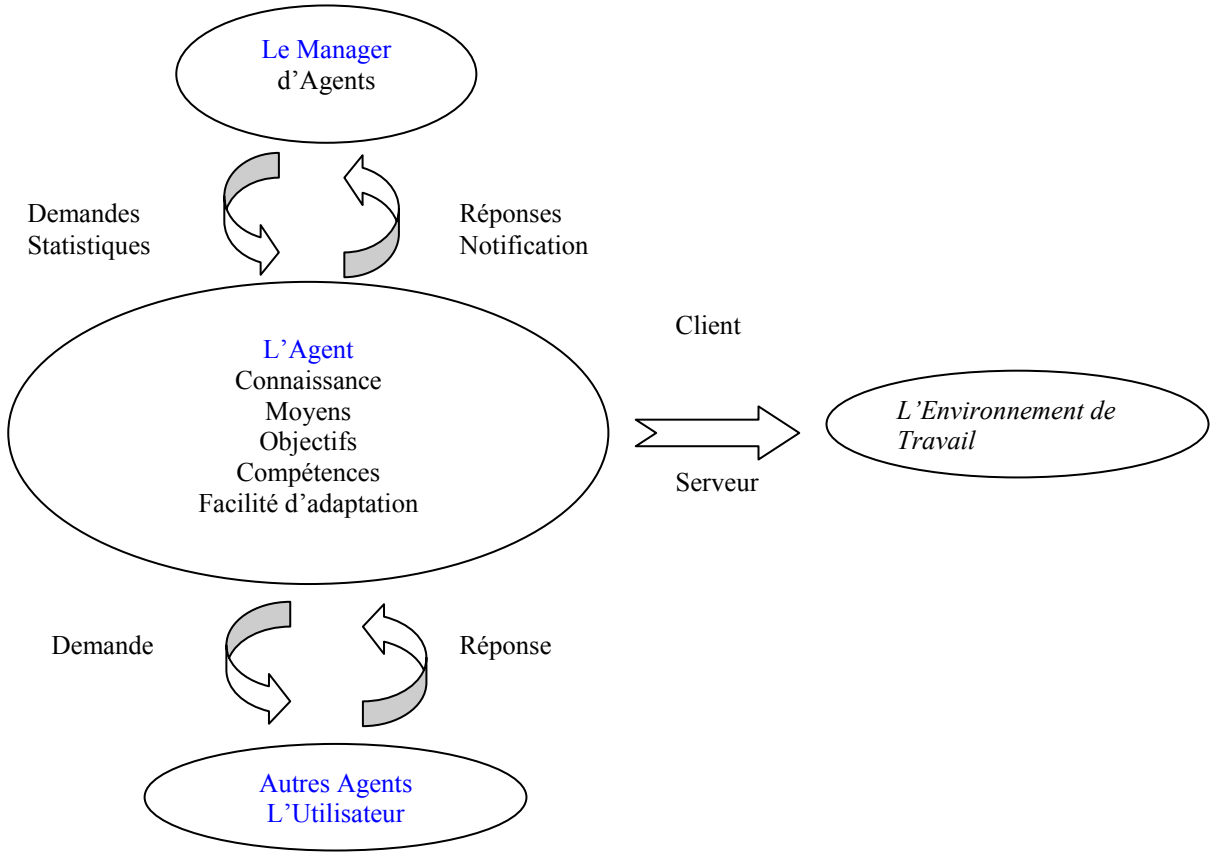


Figure 3-9 Structure d'un système multi-agents

[17] Middleware : Logiciel intermédiaire traduisant les données échangées entre plusieurs applicatifs afin de garantir l'interopérabilité des applications.

Une fois qu'il a été lancé par son processus supérieur, par un être d'humain ou un par un *Manager d'Agents*, *l'Agent Logiciel* réalise *ses objectifs* par les meilleurs *moyens* sans déranger son processus supérieur. À l'aide des scénariis et des expériences des processus courants, un Agent peut ajuster son comportement pour optimiser son environnement de travail. Ces capacités dépassent le niveau des formats prédéfinis. Elles augmentent avec chaque exécution et rajoutent de "l'intelligence" caractéristique aux Agents. C'est le *Manager d'Agents* qui définit les objectifs d'un Agent par des *statistiques* et *une configuration* donnée.

Un *Agent Logiciel* est coopératif dans le sens où il peut augmenter ses compétences par les relations avec d'autres Agents qui ont accès à d'autres ressources ou même avec un utilisateur par ses *demandes*. Il met sa propre *expérience* à la disposition d'autres Agents par des *réponses* bien précises. En coopération avec un *service* spécifique ou une application (par exemple, une base de données), un Agent agit en tant que *client* au nom d'un être d'humain ou d'un *Manager d'Agents*.

L'environnent dans lequel un Agent travaille, définit en fait l'Agent. Les Agents perçoivent leur environnent (qui peut être un endroit physique, un utilisateur par l'intermédiaire d'une interface graphique, d'autres Agents, l'Internet ou peut-être mêmes des combinaisons de ceux-ci). Un environnement précis sera défini dans notre cas.

Pour nous, l'Agent Intelligent est un outil idéal pour l'évaluation de la qualité de services multimédia. Il peut être utilisé pour contrôler, mesurer et évaluer une partie d'un réseau ainsi qu'une application précise. Une interface graphique simple et conviviale peut être aussi un atout dans son l'utilisation par un client non expert.

3.4 Une architecture d'Agent d'Evaluation de la Qualité de Service

Prenons pour exemple un Agent que nous appellerons « *Agent QoS* » pour l'évaluation de la qualité de services dans la

Figure 3-10 qui sera commentée par la suite.

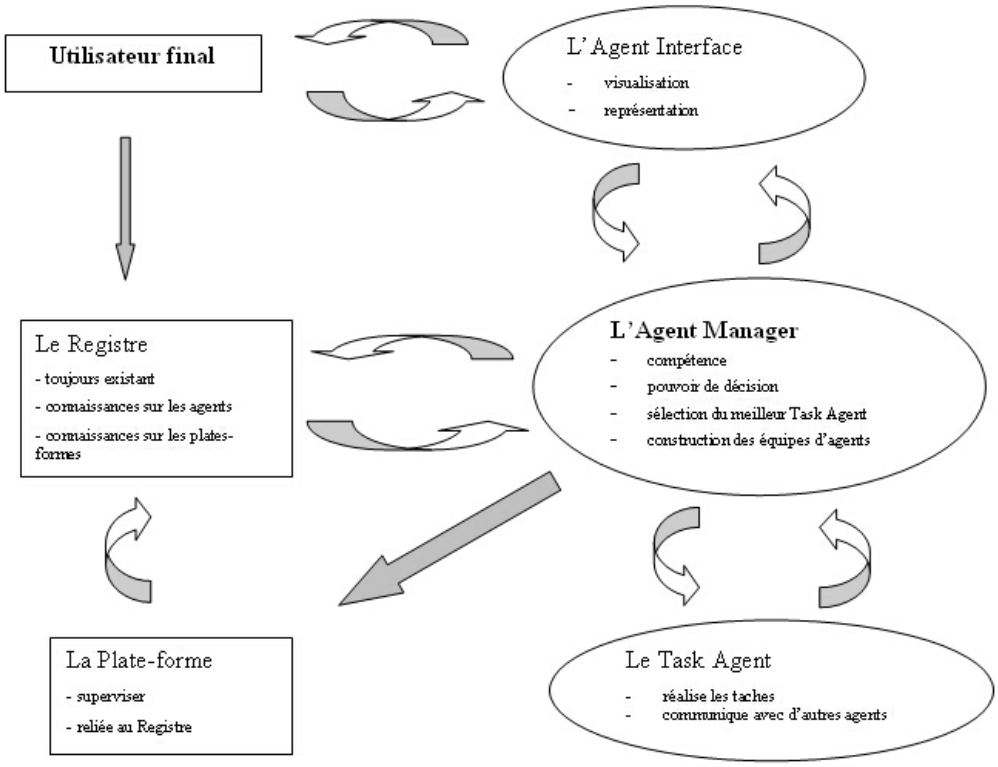


Figure 3-10 Une architecture d'Agent

L'utilisateur final qui désire évaluer certains éléments concernant la qualité de services peut rentrer en contact avec le Registre par l'intermède d'un Agent que nous appelons « *Interface* ». L'utilisateur explique sa requête en précisant le service qu'il souhaiterait évaluer et ainsi connaître son contenu. Dans *l'Agent Interface*, l'utilisateur peut trouver également des informations spécifiques sur les services qui peuvent être traités, sur les méthodes d'évaluation disponibles et des rapports des autres utilisateurs. A l'aide de cet Agent, il peut parcourir les différentes étapes de la procédure d'évaluation traitées dans le chapitre précèdent.

L'Agent Interface peut accéder aux clauses de garanties définies dans le SLA (Service Level Agreement). Comme *l'Agent Interface* est juste responsable de l'accomplissement des ordres donnés par l'utilisateur et de la présentation des résultats, il passera les informations (les conditions, la configuration et la tâche) à son tour à un *Agent Manager* qui est son supérieur hiérarchique.

L'Agent Manager est directement lié à un *Registre* qui comprend plusieurs agents que nous appellerons *Task Agent* avec différentes caractéristiques et savoir-faire, par exemple des agents peuvent calculer le temps de connexion à un serveur, sa disponibilité.

Le Registre commande plusieurs *Plates-formes* pour les Task Agents. L'Agent Manager sélectionne le *Task Agent* approprié et construira, si nécessaire, des équipes d'agents. Chaque *Task Agent* travaille donc sur une *Plate-forme* et peut communiquer s'il y a lieu, avec d'autres agents, toujours en accord avec *l'Agent Manager*. Celui-ci peut aussi détruire un agent s'il n'est plus utilisé, cependant les ressources physiques resteront disponibles au niveau du *Registre*.

Pour exécuter les cinq étapes de l'évaluation de la QoS *(l'Identification, le Classement, la Sélection, la Quantification et l'Evaluation*) à l'aide de notre *Agent QoS*, quelques précisions au niveau de l'implémentation sont nécessaires :

- les agents ont besoin d'un environnement de travail sur différentes machines (PC, SUN, etc.…) avec différents systèmes d'opération (Linux, Unix, Win 9x/2000/NT) ;

- un langage de programmation flexible et adaptable (notre choix est Java) ;

- un langage approprié de communication entre les agents (le mail, http ou CGI (Commun Gateway Interface)) ;

Dans le prochain paragraphe, nous allons présenter le langage Java ainsi que la plate forme Java sur laquelle les agents ont été développés. Le choix du langage Java a été fait par rapport à notre orientation vers les Agents Intelligents mais également pour la facilité de programmation. En ce qui concerne les mécanismes de fonctionnement, un Agent a besoin d'une machine virtuel pour fonctionner, or Java en possède une. De plus, le langage Java est sécurisé, car c'est la JVM qui vérifie au moment de l'exécution, la fiabilité du code.

3.5 Le Langage Java

3.5.1 Caractéristiques du Langage Java

Le langage Java a été créé par la société Sun Microsystems [SUN]. L'un des nombreux objectifs de Java est d'augmenter la productivité des programmeurs. Pour cela :

- il permet le « Write once, run everywhere », « écrire une fois, utiliser partout », pour ne pas être limité aux développements pour une seule plate forme (un programme écrit en Java tournera de la même façon sur MS Windows ou sur Unix). On parle de portabilité du langage (même dans sa version compilée) ;

- il est orienté objet, pour faciliter la réutilisation et la maintenance des programmes.

De plus, il y a de nombreuses fonctionnalités (classes, méthodes et interfaces) liées aux mondes des réseaux.

Le minimum nécessaire pour développer une application en langage Java est disponible gratuitement sur le site de Sun Microsystems [SUN].

3.5.2 Mécanisme général du langage Java

Pour qu'un programme écrit en langage java puisse être exécuté sur plusieurs plates-formes, il est dans un 1er temps compilé en bytecode. Le bytecode est un code pouvant être interprété avec machine virtuel, la JVM (Java Virtual Machine). Pour chaque plate-forme, une JVM a déjà été écrite par la société Sun Microsystems [SUN] ou par divers acteurs du marché comme IBM ou Borland (nous ne parlerons pas de la version « incomplète » de Microsoft).

L'interprétation du bytecode par la machine virtuel (JVM) entraîne un ralentissement du programme. C'est pour cela, qu'il est possible pour l'utilisateur du programme Java d'utiliser un compilateur JIT (Just In Time). Celui-ci permettra à la machine virtuelle de compiler le programme bytecode en code natif du CPU (langage machine) au cours de son

exécution. Le programme ne sera plus interprété par la JVM mais par le processeur de la machine grâce à une compilation qui aura eu lieu « juste à temps » (JIT).

3.5.3 Java mode d'emploi

Il existe plusieurs types de programmes en Java dont les applets et les applications. Une application autonome (stand alone program) est une application qui s'exécute sous le contrôle direct du système d'exploitation. Une applet est une application qui est chargée par un navigateur et qui est exécutée sous le contrôle de celui ci.

Pour programmer en Java, il faut :

- un éditeur de texte pour saisir le code Java ;
- Le JDK (Java Development Kit). Ce kit, nommé JDK, est librement téléchargeable sur le site web de Sun http://java.sun.com ou par FTP ftp://java.sun.com/pub/ contient javac (le compilateur de bytecode), java (l'interpréteur de bytecode) et de la documentation. Le JDK contient, en plus des packages de définition du noyau (utilisés par tous les éditeurs du marché), une implémentation dite de référence du langage. Cette implémentation de référence est principalement illustrée par les utilitaires « javac » et « java ».

Pour développer une application en Java, il faudra y ajouter le JRE (Java Runtime Environnement) qui contient lui aussi un interpréteur de bytecode, un compilateur JIT. Le JRE contient uniquement l'environnement d'exécution de programmes Java. Le JDK contient lui même le JRE. Le JRE seul doit être installé sur les machines ou des applications java doivent être exécutées.

Depuis sa version 1.2, Java a été renommé Java 2. Les numéros de versions 1.2 et 2 désignent donc la même version.

Le JDK standard a été renommé J2SE (Java 2 Standard Edition) mais la dénomination JDK reste encore largement utilisée. Le JRE a été renommé J2RE (Java 2 Runtime Edition).

Trois éditions de la plate-forme Java 2 existent :

- Java 2 Micro Edition (J2ME) : contient le nécessaire pour développer des applications capables de fonctionner dans des environnements limités tels que les assistants personnels (PDA), les téléphones portables ou les systèmes de navigation embarqués (une présentation plus en détails sera faite dans le chapitre suivant) ;

- Java 2 Standard Edition (J2SE) : contient le nécessaire pour développer des applications et des applets. Cette édition reprend le JDK 1.0 et 1.1 ;

- Java 2 Enterprise Edition (J2EE) : contient en plus un ensemble de plusieurs API permettant le développement d'applications destinées aux entreprises telles que JDBC pour l'accès aux bases de données, EJB pour développer des composants orientés métiers, Servlet / JSP pour générer des pages HTML dynamiques.

Le but de ces trois éditions est de proposer une solution à base Java quelque soit le type de développement à réaliser. Sun fournit le JDK 1.2 sous les plate-formes Windows, Solaris et Linux.

Sun propose aussi la documentation en format HTML des API de Java qui est fournie séparément. Malgré sa taille imposante, cette documentation est indispensable pour obtenir des informations complètes sur les classes utilisées.

3.5.4 Le Java 2 Micro Edition (J2ME)

Nous nous sommes intéressés à cette édition de la plate-forme Java 2 puisqu'elle correspond au mieux à notre approche vers la téléphonie mobile et la possibilité de l'évaluation de la qualité de service à partir d'un terminal mobile. Créée par Sun, la technologie J2ME (Java 2 Micro Edition) permet aux développeurs de concevoir des applications embarquées destinées à un large éventail d'appareils portables comme l'assistant personnel (PDA) classique (voir Figure 3-11).

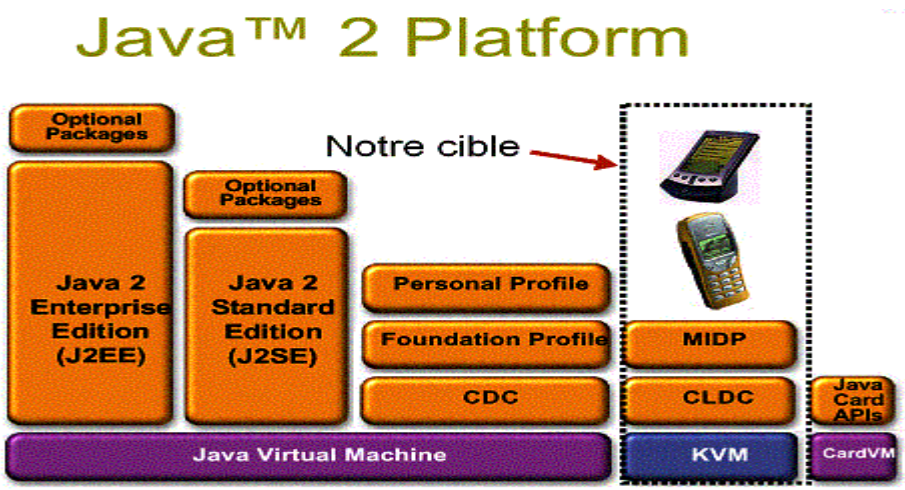


Figure 3-11 Architecture Plateforme Java 2 Micro Edition

Les terminaux n'ayant pas les mêmes capacités en terme de ressources que les ordinateurs de bureau classiques (mémoire, disque et puissance de calcul), la solution passe par la fourniture d'un environnement allégé afin de s'adapter aux différentes contraintes d'exécution. Cependant, comment faire en sorte d'intégrer la diversité de l'offre à un socle technique dont la cible n'est pas définie à priori ? La solution proposée par J2ME consiste à regrouper par catégories certaines familles de produits tout en proposant la possibilité d'implémenter des routines spécifiques à un terminal donné. Les points forts de cette solution portent sur la richesse de l'interface utilisateur et sur le fonctionnement, au choix, en mode connecté ou en mode déconnecté.

L'architecture J2ME se découpe donc en plusieurs couches [Réf. DotNet] avec différentes contraintes techniques :

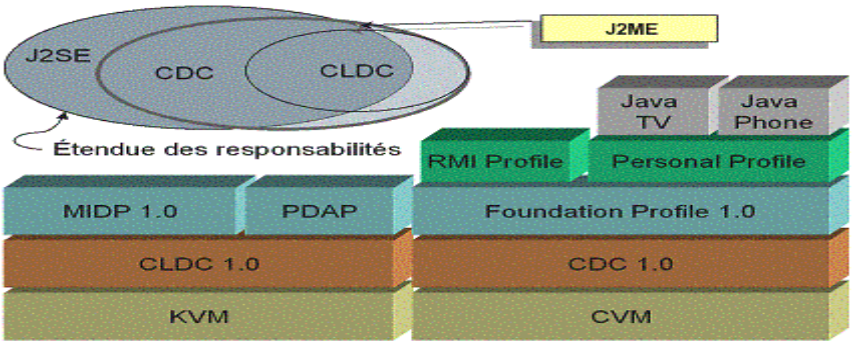


Figure 3-12 Différentes couches de J2ME

Cette architecture en couches (Figure 3-12) a pour but de factoriser pour des familles de produits données un ensemble d'API permettant à une application de s'exécuter sur plusieurs terminaux sans modification de code.

Les briques de base de J2ME sont la configuration, le profil et les packages optionnels [Réf Delb].

- une configuration est une machine virtuelle et un ensemble minimal de classes de base et d'API. Elle spécifie un environnement d'exécution généralisé pour les terminaux embarqués et agit comme plate-forme Java sur le terminal ;

- un profil est une spécification des API Java définie par l'industrie et utilisée par les fabricants et les développeurs à destination des différents types de terminaux possibles ;

- un package optionnel est comme son nom l'indique, un package qui peut ne pas être implémenté sur un terminal particulier.

Les machines virtuelles

La machine virtuelle se trouve entre l'application et la plate-forme utilisée, convertissant les bytescode de l'application en mode machine approprié au matériel et au système d'exploitation utilisé. En fonction de la cible, la machine virtuelle pourra être allégée afin de consommer plus ou moins de ressources. J2ME propose aujourd'hui 2 types de machines virtuelles, **KVM** (KiloByte Virtual Machine) et **CVM** (Convergence Virtual Machine).

Les classes Java s'exécutant dans la **KVM,** ont été conçues pour fonctionner dans un environnement limité en terme de mémoire (128 K), d'énergie et d'accès réseau. Cette configuration s'inscrit donc dans une logique d'économie de ressources avec une KVM de 40 à 80 Ko s'exécutant 30 à 80% moins vite qu'une JVM normale.

Concernant la CVM, elle a été conçue pour les terminaux ayant besoin de l'ensemble de fonctionnalités de la JVM mais avec des capacités plus réduites. Les terminaux utilisant CVM sont généralement des terminaux compacts et connectés, orientés consommateur.

Les configurations

Elles définissent une plate-forme minimale en terme de services concernant un ou plusieurs profils donnés. Deux configurations sont proposées aujourd'hui : le **CDC (Connected Device Configuration)** et **CLDC (Connected Limited Device Configuration).**

- le **CDC** spécifie un environnement pour des terminaux connectés de forte capacité tels que les « set top boxes », les téléphones à écran, la télévision numérique. Les caractéristiques de l'environnement matériel proposé par la configuration CDC sont :

- un minimum de **512Ko** de ROM et **256Ko** de RAM, processeur **32 bits** ;
- une connexion réseau obligatoire (sans fil ou pas) ;
- un support des spécifications complètes de la machine virtuelle Java (CVM).

Cette configuration s'inscrit donc dans le cadre d'une architecture Java presque complète.

- le **CLDC** cible les périphériques à ressources limitées ou faibles tels que les téléphones mobiles, les assistants personnels, ou les périphériques légers sans fil (wireless). Ces périphériques étant limités en terme de ressources, l'environnement classique ne permet pas de respecter les contraintes d'occupation mémoire liées à ces appareils. J2ME définit donc un ensemble d'API spécifique à CLDC et destiné à utiliser les particularités de chaque terminal d'une même famille (profile). Les caractéristiques de l'environnement matériel proposé par la configuration CDLC sont :

- un minimum de **160Ko à 512Ko** de RAM, processeur **16 ou 32 bits,** vitesse **16Mhz** ou plus ;
- **une alimentation limitée**, prise en charge d'une batterie ;
- une connexion réseau non permanente, sans fil ;
- une interface graphique limitée ou inexistante.

Toutefois, CLDC n'intègre pas la gestion des interfaces graphiques, la persistance ou les particularités de chaque terminal. Ces aspects ne sont pas de sa responsabilité. Le Tableau 3-4 résume les packages et classes présentes dans cette couche :

Liste des packages de CLDC	
Java.io	Fournit la gestion des flux d'entrées/sorties
Java.lang	Classes de base du langage (types, …)
Java.util	Contient les collections et classes utilitaires
Javax.microedition.io	Classes permettant de se connecter via TCP/IP

Tableau 3-4 Liste de packages CDLC

Les profils

Ils permettent à une certaine catégorie de terminaux d'utiliser des caractéristiques communes telles que la gestion de l'affichage, des évènements d'entrées/sorties (pointage, clavier, …) ou des mécanismes de persistance (Base de données légère intégrée). Ces profils sont soumis à spécifications suivant le principe du **JCP** (Java Community Process). Les profils définissent l'ensemble complet de classes API qui seront mises à disposition d'une application J2ME et conçues spécifiquement pour une configuration donnée (Figure 3-13).

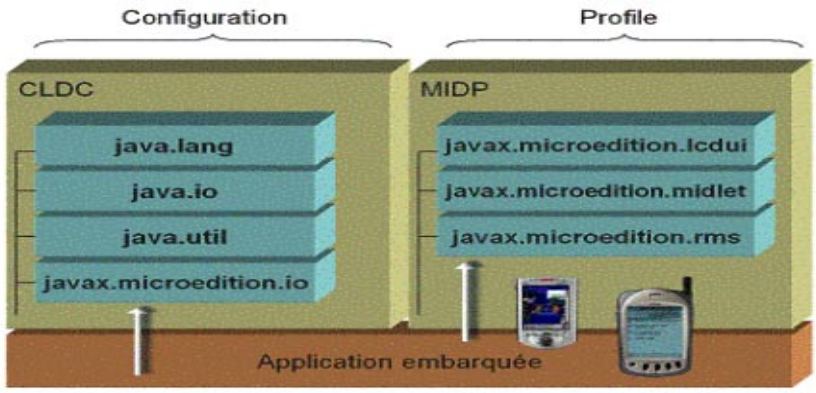


Figure 3-13 Configuration, Profile J2ME

Sun propose deux profils de référence J2ME : le profil Foundation et le profil Mobile Information Device Profile (MIDP).

Le profil Foundation est destiné à la configuration CDC et offre par conséquent toute la richesse d'une machine virtuelle quasiment identique à la machine virtuelle standard. Cela signifie que les développeurs qui utilisent le profil Foundation ont accès à une implémentation complète des fonctionnalités de J2SE.

Le profil MIDP est un profil pour les appareils mobiles utilisant la configuration CLDC, comme les téléphones mobiles actuels. Le profil MIDP précise les fonctionnalités comme l'usage de l'interface client, la persistance de stockage, la mise en réseau, et l'application modèle. Il prend en charge un sous-ensemble limité de la bibliothèque de classes J2SE et définit des classes d'entrée/sortie (saisie utilisateur) et d'interface utilisateur personnalisée pour une utilisation sur une configuration CLDC. Il existe aujourd'hui deux implémentations majeures de profils MIDP :

- une spécifique [Réf. Midp4Palm] destinée aux Assistants de type Palm Pilot (PalmOs)

- une générique, proposée par Sun comme implémentation de référence (RI) [Réf. J2MEtoolkit].

Ces profils sont en libre téléchargement sur le site de Sun et intègrent plusieurs émulateurs permettant de tester les applications de manière logicielle. Les API liées à MIDP sont présentées dans le tableau Tableau 3-5.

Liste des packages MIDP	
javax.microedition.lcdui	Fournit la gestion de l'interface utilisateur (contrôles, …)
javax.microedition.midlet	Socle technique destiné à gérer le cycle de vie des Midlets[18]
javax.microedition.rms	Base de données persistante légère

Tableau 3-5 Les packages MIDP

J2ME propose toute une palette d'outils à travers le J2ME Toolkit. Tout d'abord, un Runner (voir Figure 3-14) d'applications MIDP, c'est à dire un programme destiné à lancer les exécutables (fichiers jad) créés par le développeur tout en proposant une série de 6 émulateurs allant du Palm Pilot classique aux téléphones mobiles Motorola i85. L'idée de cette série d'émulateurs est de démontrer que la même application développée pour un profil donné pourra s'exécuter sur plusieurs périphériques sans modification préalable.

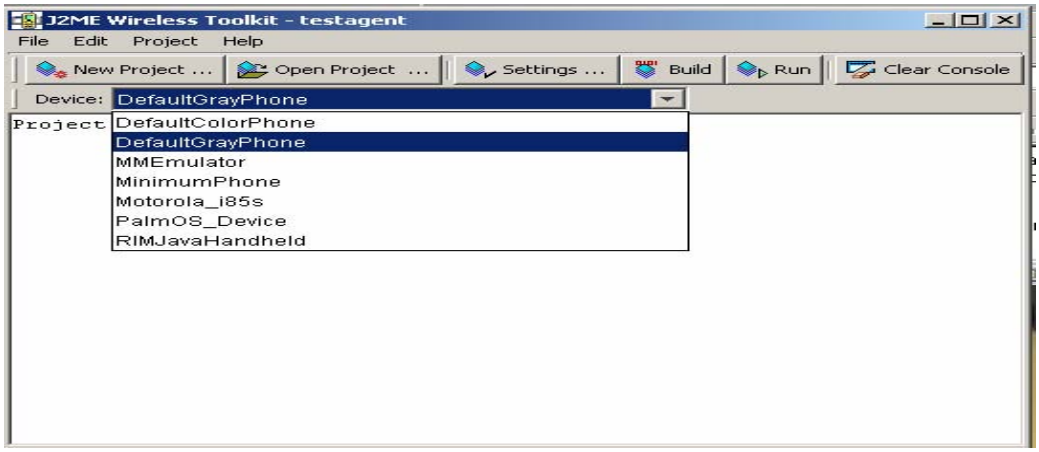


Figure 3-14 Wireless Sim Toolkit

Toutes nos simulations ont été réalisées avec l'émulateur DefaultGreyPhone (Figure 3-15).

[18] MIDLET : élément principal d'une application Java embarquée. En cas de mise à jour d'une application embarquée, un simple téléchargement de code Midlet est nécessaire à partir d'un quelconque serveur.

Figure 3-15 Le DefaultGreyPhone

Une présentation de chaque étape de la simulation sera faite lors de la Section 5.

3.6 Résumé du chapitre

Dans ce chapitre, nous avons présenté dans un premier temps le modèle d'un mécanisme pour l'évaluation de la qualité de service. Les 4 principales parties de ce modèle sont un Utilisateur, un Registre, une Implémentation ainsi qu'une Application/Service pour tester. Par rapport à notre modèle, nous avons établi un procédé simple pour l'utilisateur final concernant l'évaluation de la qualité de service. Ce procédé en 5 pas est basé en autre sur le contrat de garantie de service (SLA – Service Level Agreement) établi entre l'utilisateur final et son opérateur ou fournisseur de service.

Une architecture de l'évaluation de la QoS à l'aide des Agents Intelligents ainsi que le choix en terme de langage de programmation des Agents (Java) représente la dernière partie de ce chapitre.

Dans le prochain chapitre, nous allons préciser comment la procédure de l'évaluation de la qualité de service en 5 étapes a été réalisé pour les 3 services multimédia répartis : le service de courrier électronique, les services web en ligne (www) et les services de streaming vidéo MPEG-1.

4 Application de la procédure d'évaluation de la qualité de service

Dans le chapitre 4 , nous allons appliquer la procédure d'évaluation de la qualité de service présentée dans le chapitre 3, pour 3 services multimédia repartis : le service de courrier électronique, les services web en ligne (www) et les services de streaming vidéo MPEG-1. Afin de prouver que notre processus d'évaluation est fiable, les caractéristiques, les normes, et les recommandations différentes peuvent être employées en tant que point de départ. Les exemples numériques donnés, basés sur la seule expérience de l'auteur, servent seulement d'illustration. C'est l'utilisateur ou le fournisseur de service qui, finalement, doit définir selon différentes conditions et contraintes les paramètres de la qualité qu'il souhaite tester.

4.1 Evaluation de la Qualité de Service pour le courrier électronique

Le Courrier électronique (Mél ou e-mail) est l'un des outils Internet les plus populaires aujourd'hui. Nous estimons qu'environ 150 millions d'utilisateurs l'emploient en Europe.

De la même manière que dans le courrier postal, il permet à deux ou plusieurs personnes de s'envoyer des messages. (La différence étant que les messages sont sous forme numérique et qu'on utilise son ordinateur comme boîte postale).

C'est un moyen de communication rapide, peu coûteux (prix de la connexion Internet), qui permet, de plus, d'envoyer des documents, des sons et des images en format numérique.

Le courrier électronique fonctionne selon le mode client/serveur (Figure 4-1). SMTP[19] est le principal protocole Internet pour envoyer et recevoir des mails.

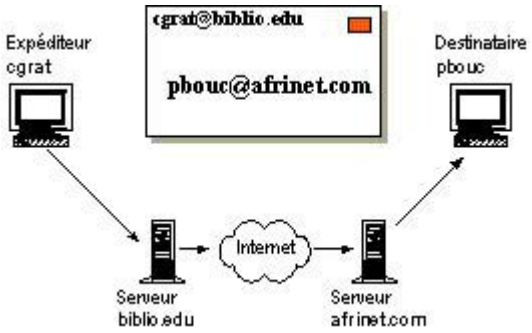


Figure 4-1 Acheminement du courrier électronique

Le courrier électronique utilise le même principe d'adresse que le courrier postal.

Le mode d'adressage est composé de deux parties reliées entre elles par le symbole @ :

- un nom d'usager qui correspond à un casier postal,
- un nom de domaine qui identifie le bureau de poste qui abrite les casiers postaux.

Le logiciel de courrier électronique client est utilisé par l'usager pour :

- rédiger un message électronique,
- envoyer et recevoir du courrier électronique,
- lire les messages reçus,

Le logiciel serveur est utilisé par le serveur pour :

- entreposer les messages reçus. Ainsi, lorsqu'un usager va chercher ses messages, il se connecte à son serveur de courrier électronique pour les télécharger;
- acheminer les messages envoyés par l'usager sur Internet jusqu'à leur destination finale, c'est-à-dire jusqu'au serveur de courrier électronique du destinataire.

Netscape, Pegasus, Exchange, Eudora, Outlook et Internet Mail sont des exemples de logiciels client de courrier électronique.

Le courrier électronique offre des avantages par rapport à d'autres équipements de communication en termes de fiabilité et facilité d'utilisation. De plus, tous les utilisateurs d'email tirent bénéfice des dispositifs tels que :

- la communication à travers le monde ;
- la communication à travers une ou plusieurs listes de distribution ;

---

[19] SMPT : Simple Mail Transport Protocol

- la possibilité de transfert de messages ;
- la préparation soigneuse d'un message ;
- la facilité de lecture ;
- la possibilités d'attacher des fichiers multimédia (acoustique, vidéo, graphiques, voix,...),
- l'administration facile des messages envoyés et reçus (archives).

Les normes les plus importantes pour le courrier électronique sont [Réf.1521] et [X.400/88]. La référence [Réf. 1521] décrit le standard MIME (Multipurpose Internet Mail Extensions) proposé par l'IETF en 1992. Ce standard met en valeur la capacité du mail à transmettre via Internet différents types de média :

- Des textes en format ASCII[20] ;
- Des caractères autres que ASCII ;
- Des messages qui contiennent des fichiers multimédia: Image, Audio, et Vidéo ;
- Plusieurs objets dans un seul message ;
- Des messages multi fontes ;
- Des messages avec une taille illimitée (il existe quand même une limite de taille du mail qui dépend de la configuration du serveur mail) ;
- Des fichiers binaires.

Le standard MIME a été développé pour être compatible avec tout autre type de messages. Il est flexible et ouvert vers d'autres extensions (par exemple, en rajoutant de nouveaux champs dans l'en-tête du mail pour décrire des nouveaux types de contenu du mail et une disposition à part du corps du message). Les buts de ces perfectionnements sont :

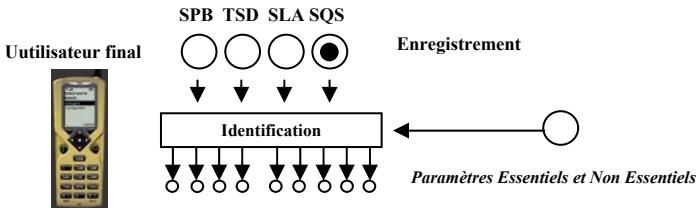
[20] ASCII : American Standard Code for Information Interchange

- de diffuser l'information de multimédia efficacement,
- de permettre la transmission par morceaux des corps de messages supposés trop grands pour être transmis par les équipements de transport de courrier [Réf1521],
- d'indiquer de grands corps en se référant à une source extérieure de données [Réf1521].

Par rapport à MIME dans les normes [X.400/84] et [X.400/88], on trouve peu d'informations sur les pièces attachées de type multimédia. Les définitions précises manquent ("for further study ") ou des définitions toujours externes sont prises ("externally-defined body part ") pour des objets tels que la voix, l'audio, la vidéo, les documents composés et les images. Pour des informations complètes concernant les standards de mail voir la [Réf. ASG].

Beaucoup de compagnies et d'organismes ont un problème pour trouver un fournisseur de service email et pour choisir le logiciel de courrier électronique approprié à leurs besoins spécifiques. Même si le mail en lui-même est assez simple, les systèmes de courrier électronique aujourd'hui deviennent de plus en plus complexes. Il est de plus en plus difficile d'installer des système de courrier performants. Dans cette idée, nous avons essayé de construire un outil simple pour évaluer la qualité de service pour les mails.

4.1.1 Identification de tous les paramètres de la qualité



Dans ce premier pas, tous les paramètres essentiels et non essentiels de la qualité de service pour les mails doivent être identifiés. Les normes dont nous avons parlé ne donnent aucune information concernant la qualité de service pour les mails. En conséquence, nous allons utilisé le standard ISO 9126 [Réf ISO 9126] en tant que standard logiciel sur lequel nous fondons notre évaluation. Dans cette démarche, il est important aussi de tenir compte de la SLA (Service Level Agreement) dans lequel nous

trouvons des informations concernant l'utilisateur final et son accès à l'utilisation des mails.

Nous pouvons identifier les caractéristiques du logiciel à partir de la norme ISO 9126. Une définition de celle-ci ainsi que ses sous-caractéristiques seront présentes dans le Tableau 4-1.

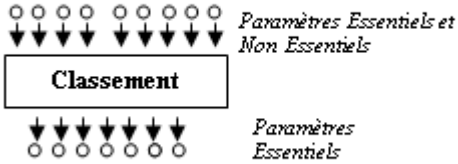
Paramètres	**Description**	**Sous-paramètres**
Capacité fonctionnelle	Ensemble d'attributs portant sur l'existence d'un ensemble de fonctions et leurs propriétés données. Les fonctions sont celles qui satisfont aux besoins exprimés ou implicites.	Aptitude à la tâche, Précision, Interopérabilité, Conformité aux normes, Sécurité d'accès
Fiabilité	Ensemble d'attributs portant sur l'aptitude du logiciel à maintenir son niveau de service dans des conditions précises et pendant une période déterminée.	Maturité: Fréquence des défaillances (MTF), Densité des défauts, Stabilité du produit, Tolérance aux défaillances, Densité et couverture des tests, Facilité de recouvrement en cas de défaillance
Facilité d'utilisation	Ensemble d'attributs portant sur l'effort nécessaire pour l'utilisation et sur l'évaluation individuelle de cette utilisation par un ensemble défini ou implicite d'utilisateurs.	Facilité de compréhension, Facilité d'opération, Facilité d'apprentissage
Rendement	Ensemble d'attributs portant sur le rapport existant entre le niveau de service d'un logiciel et la quantité de ressources utilisées, dans des conditions déterminées.	Par rapport au temps: Réponse, UCT[21], E/S, Attente, Réseau ; Par rapport aux ressources (taille et utilisation): Mémoire principale, Mémoire virtuelle, Fichiers, Réseau
Facilité d'entretien	Ensemble d'attributs portant sur l'effort nécessaire pour faire des modifications données.	Par rapport à l'analyse des défaillances, Par rapport aux modifications, Stabilité (risque d'effets imprévus après modification) Facilité de validation
Portabilité	Ensemble d'attributs portant sur l'aptitude de logiciel à être transféré d'un environnement à l'autre.	Facilité d'adaptation à différents environnements, Facilité d'installation,degré d'ouverture, Facilité de substitution pour un autre logiciel similaire

Tableau 4-1 Caractéristiques du logiciel ISO 9126

Cette étape nous a permis de situer l'ensemble de paramètres essentiels et non essentiels (voir le Tableau 4-1) pour l'évaluation de la qualité de service pour les mails.

[21] UCT : Unité Centrale de Traitement. Souvent, c'est simplement le processeur.

4.1.2 Etape 2: Le classement



Au cours de cette étape les paramètres essentiels doivent être séparés de l'ensemble des paramètres trouvés au cours de la première étape. En tenant compte de la présence de l'utilisateur final dans notre évaluation et du fait que les moyens techniques au niveau de l'Implémentation sont limités, on s'est arrêté sur 3 paramètres parmi les 6 présentés. *La capacité fonctionnelle, la fiabilité et le rendement,* nous semblent les paramètres les plus proches de l'utilisateur final. Les paramètres comme *la facilité d'entretien et la portabilité* ont été éliminés étant des paramètres internes au fournisseur de service. La facilité de l'utilisation a été éliminée tout simplement parce que l'utilisateur final a besoin de tester la QoS qui lui est proposée et pas du logiciel, dont il se sert pour le tester.

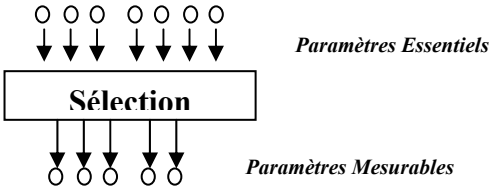
Etant donné l'impossibilité de la mise en place dans notre implémentation de toutes les sous caractéristiques sur les 3 paramètres essentiels retenus en ce qui concerne l'évaluation de la QoS pour le mail, on a retenu les suivants (voir Tableau 4-2) :

Paramètres	**Sous paramètres**	**Description**
Capacité fonctionnelle	Aptitude à la tâche	La présence d'un ensemble de fonctions appropriées pour des tâches indiquée (c.-à-d., conformité X.400 ou SMTP).
	Précision	La disposition et l'exactitude des résultats fournis.
	Interopérabilité	La capacité de collaborer avec les systèmes indiqués (c.-à-d., X.400, SMTP, UUCP-Mail).
	Conformité aux normes	Conformité à l'application des normes ou des conventions ou des règlements dans les lois (ex X.400).
	Sécurité d'accès	Empêcher l'accès non autorisé, accidentel ou délibéré à des programmes et des données.
Fiabilité	Maturité: Fréquence des défaillances	Maintenir un niveau indiqué d'exécution dans le cas de défauts du logiciel ou des problèmes au niveau de l'interface indiquée.
	Facilité de recouvrement en cas de défaillance	Le rétablissement de son niveau d'exécution et la récupération des données directement affectées en cas d'échec.
Rendement	Par rapport aux ressources (taille et utilisation): Mémoire principale	La mémoire tampon utilisée en cas d'envoi de fichiers attachés.

Tableau 4-2 Les paramètres essentiels du service mail

Après cette étape, les paramètres et les sous-paramètres essentiels pour l'évaluation de la QoS pour le mail ont été déterminés.

4.1.3 Etape 3: Choix de mesures



Dans cette étape un choix doit être fait sur les paramètres essentiels qui peuvent être mesurés d'une manière automatique. Encore une fois, la participation de l'utilisateur final est importante. Même si sa participation à cette étape n'est pas directe, son point de vue concernant les paramètres à tester concernant les services proposés est déterminant et les opérateurs doivent en tenir compte.

Dans un réseau mondial avec beaucoup de produits et systèmes différents, il est important d'avoir des normes, qui décrivent la manière de communiquer entre deux produits différents.

L'interopérabilité des systèmes répartis hétérogènes et la conformité avec les normes internationales peuvent être considérées comme des propriétés fonctionnelles d'un service qui sont trop compliquées pour que les utilisateurs les mesurent. Elles sont estimées comme garanties avec l'abonnement à un service. Cependant, elles doivent être considérées comme caractéristiques de la qualité ; leur évaluation doit être faite avant le lancement du service par des entités compétentes.

Cet argument nous conduit aux paramètres de la qualité à mesurer (Tableau 4-3).

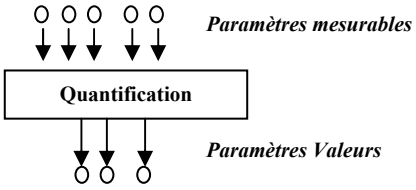
Paramètres	Sous paramètres	Attributs
Capacité fonctionnelle	Précision	Domaine, Alias, messages envoyés à l'extérieur, messages reçus de l'extérieur, dernier message envoyé, dernier message reçu, dernier login
Fiabilité	Maturité: Fréquence des défaillances	Nombre de messages perdus par rapport au nombre de messages reçus Nombre de messages perdus par rapport au nombre de messages envoyés
	Facilité de recouvrement en cas de défaillance	Nombre de messages récupérés en cas de défaillance.
Rendement	Par rapport aux ressources (taille et utilisation): Mémoire principale	Quantité Envoyée, Quantité Reçue

Tableau 4-3 Paramètres essentiels de la qualité de service à mesurer pour le mail

Des paramètres comme « l'aptitude à la tâche », « l'interopérabilité » et « la conformité aux normes » ont été éliminés pour les raisons décrites auparavant. « La sécurité d'accès à des programmes et des données» est un paramètre qui concerne plus les fournisseurs de service que l'utilisateur final (dans le contrat de service, il existe des clauses concernant la sécurité d'accès aux services proposées par l'opérateur mobile ou le fournisseur de service).

Les attributs que nous avons proposés sont en fait des paramètres qu'on trouve sur les fichiers **.log** du serveur mail.

4.1.4 Etape 4: Quantification des Paramètres mesurables



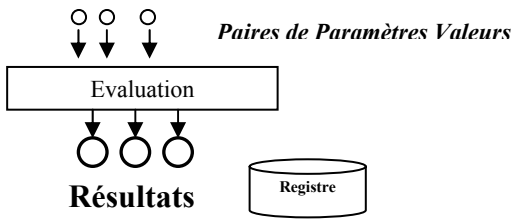
Dans cette étape, une valeur numérique doit être attribuée pour chaque paramètre mesurable. Dans le cas du mail, cette valeur numérique n'est pas fixe, elle dépend de la fréquence d'utilisation du service. Dans notre cas, les exemples de valeurs que nous avons proposées sont basés sur l'expérience de l'auteur en matière d'utilisation des mails et dans la littérature spécialisée [Réf Bogen00].

Paramètres	Sous paramètres	Attributs	Valeurs
Capacité fonctionnelle	Précision	Domaine Alias Messages envoyés à l'extérieur Messages reçus de l'extérieur Dernier message envoyé Dernier message reçu Dernier login	testevry.int-evry.fr augustin X X Date Date augustin
Fiabilité	Maturité: Fréquence des défaillances	Nombre de messages perdus par rapport au nombre de messages reçus Nombre de messages perdus par rapport au nombre de messages envoyés	< 3% < 3%
	Facilité de recouvrement en cas de défaillance	Nombre de messages récupères en cas de défaillance.	>97%
Rendement	Par rapport aux ressources (taille et utilisation): Mémoire principale	Quantité Envoyé Quantité Reçue	X X

Tableau 4-4 Paramètre valeur pour le mail

Dans notre tableau la valeur « X » veut dire un nombre entier qui n'est pas fixe et qui dépend de l'utilisation du mail de chaque personne.

4.1.5 Etape 5: L'évaluation de la qualité de service



Cette dernière étape nous permet d'avoir des résultats concrets en ce que concerne l'évaluation de la qualité de service pour les mails.

Mesurer les paramètres de la qualité signifie dans ce cas comparer ces paramètres avec ceux que nous avons définis dans le Tableau 4-4.

A travers cette comparaison, nous allons trouver des résultats qui prouvent que :

- l'utilisateur final a été pris en compte pour l'évaluation de la qualité de service ;
- les paramètres pris en compte sont directement observables et mesurables au point où l'utilisateur final y accède ;
- sont reproductibles et comparables.

Une implémentation de cette évaluation de la qualité de service à l'aide des Agents Intelligents sera présentée dans la section 5.3.

4.2 Evaluation de la Qualité de Service pour les services en ligne (www)

L'outil qui a rendu populaire Internet, est incontestablement le WWW, le World Wide Web en un mot le Web. Le mot Web désigne en anglais la *Toile d'araignée mondiale* des serveurs d'information qui est fondée sur un ensemble de protocoles, de structures de fichiers et de syntaxes de descriptions de contenus hypermédias. Le WWW ou W3, a été créé en 1991 par le laboratoire des hautes énergies, au CERN à Genève. Il était initialement destiné à fournir un mode convivial pour échanger l'information dans cette communauté de chercheurs. Sa définition officielle était : *wide-area hypermedia information retrieval initiative aiming to give universe an access to a large universe of documents*. Très rapidement, la technologie Web s'est répandue, d'abord dans le monde de la physique nucléaire, puis dans l'ensemble des domaines de la recherche scientifique, et enfin, avec l'ouverture d'Internet aux usages non académiques. Le projet W3 a donné aux utilisateurs d'Internet, un outil efficace pour accéder à une grande variété de documents de façon très simple. Grâce à des interfaces clientes conviviales, le projet W3 a changé la façon de voir et de créer l'information des utilisateurs. Parmi les termes couramment utilisés en ce qui concerne le web, on trouve donc le ***WWW*** (World Wide Web), le ***URL*** (Uniform Ressource Locator), le ***DTP*** (Document Transfer Protocol), ***HTML*** (Hypertext

Markup Language), ***XML*** (Extensible Markup Language) ou tout simplement le mot « document ». Tous ces termes seront définis par la suite.

- le ***W**orld **W**ide **W**eb* : une toile d'araignée de serveurs d'informations reliés les uns aux autres par des liens physiques (le réseau matériel) et des liens logiques (les liens hypertextes). Ces liens hypertextes permettent de voyager d'un serveur à l'autre sur le réseau Internet ;

- le ***U**niform **R**essource **L**ocator* est une extension au niveau d'Internet de la notion de nom de fichier sur une machine qui permet d'adresser de façon précise toute ressource accessible sur Internet ;

- le ***D**ocument **T**ransfer **P**rotocol* est un protocole utilisé avec le WWW pour invoquer des actions sur des différents documents. Parmi les protocoles les plus connus on trouve le HTTP (Hypertext Transfer Protocol) qui est un protocole regroupant les conventions des transmissions de pages via Internet et le FTP (File Transfer Protocol) qui est un protocole Internet permettant aux utilisateurs d'échanger des fichiers entre ordinateurs ;

- le ***H**ypertext **M**arkup **L**anguage* est le langage du World Wide Web décrivant l'affichage et l'exécution des données sur les navigateurs ;

- Le ***E**xtensible **M**arkup **L**anguage* d'un standard issu du Consortium W3 [Réf W3], le groupe chargé de définir les normes du Web. Comme le HTML comporte un nombre limité de marqueurs, à l'aide du XML, l'utilisateur peut définir ses propres marqueurs. D'autre part, le XML présente d'importants avantages pour l'édition Web comme la facilité pour définir la structure des documents. Néanmoins, le HTML est là pour rester. Dans un futur proche il est prévu d'utiliser la combinaison des deux technologies: le XML pour structurer l'information et le HTML pour déterminer la façon dont les données s'afficheront dans le navigateur représente une

- le « document » est un ensemble cohérent de différents types de données (texte, image, vidéo) réalisent un HTML par exemple et accessible sur le WWW par un URL ou un DTP.

Aujourd'hui le WWW et l'Internet se sont assimilés et sont devenus une source inépuisable d'informations. Avec l'arrivé de réseaux mobiles de 3ème génération nous sommes sur le chemin de la convergence fixe-mobile. Si au niveau du contenu, les choses ont pas mal avancé (on trouve des applications de l'Internet classique sur les téléphones mobiles), l'accès reste un point essentiel et sensible pour un utilisateur « fixe » ainsi que « mobile ». Le choix du fournisseur d'accès est une décision d'ordre stratégique pour une société et d'ordre financier pour un utilisateur final. Donc notre évaluation de la qualité de service sera basée au niveau de l'accès à un service www. La [Réf. 2594] justifie le besoin de l'évaluation de la qualité de service comme suit :

The usability of the World Wide Web depends largely on the performance of the services realized by these servers. The services are typically monitored through log files. This becomes a difficult task when a single organization is responsible for a large number of services. It is therefore desirable to treat WWW services as objects that can be managed by using the Internet network management framework [RFC2039].

L'évaluation de la qualité de service concernant les réseaux Internet contient plusieurs facettes que nous pouvons classés en 3 groupes :

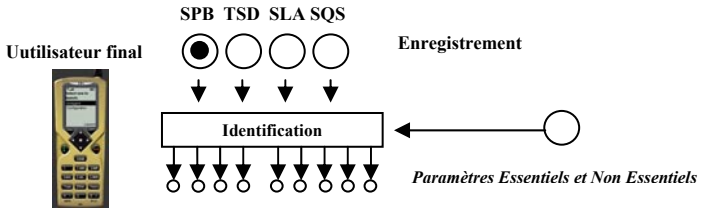
- groupe 1 : les utilisateurs ou consommateurs de services. Il comporte l'accessibilité et la connexion aux applications, l'accès aux serveurs de noms ou la perception du réseau ressentie par l'utilisateur ;

- groupe 2 : les serveurs. Il comporte la connexion, la gestion et la distribution des applications ;

- groupe 3 : les machines de transport. Il correspond aux opérations, les incidents, les performances, les coûts, la configuration, l'évolution et les changements sur le réseau.

Notre but est de trouver dans ces 3 groupes les paramètres significatifs pour l'utilisateur en ce qui concerne l'évaluation de la QoS.

4.2.1 Identification de tous les paramètres de la qualité



Dans ce premier pas, tous les paramètres essentiels et non essentiels de la qualité de service pour les services en ligne de types « www » doivent être identifiés. Ces paramètres représentent des informations statistiques concernant un serveur qui peuvent être statiques (un fichier) ou dynamiques (les résultats de plusieurs mesures). Pour mieux comprendre la différence entre les 2 types d'informations, nous vous proposons la Figure 4-2.

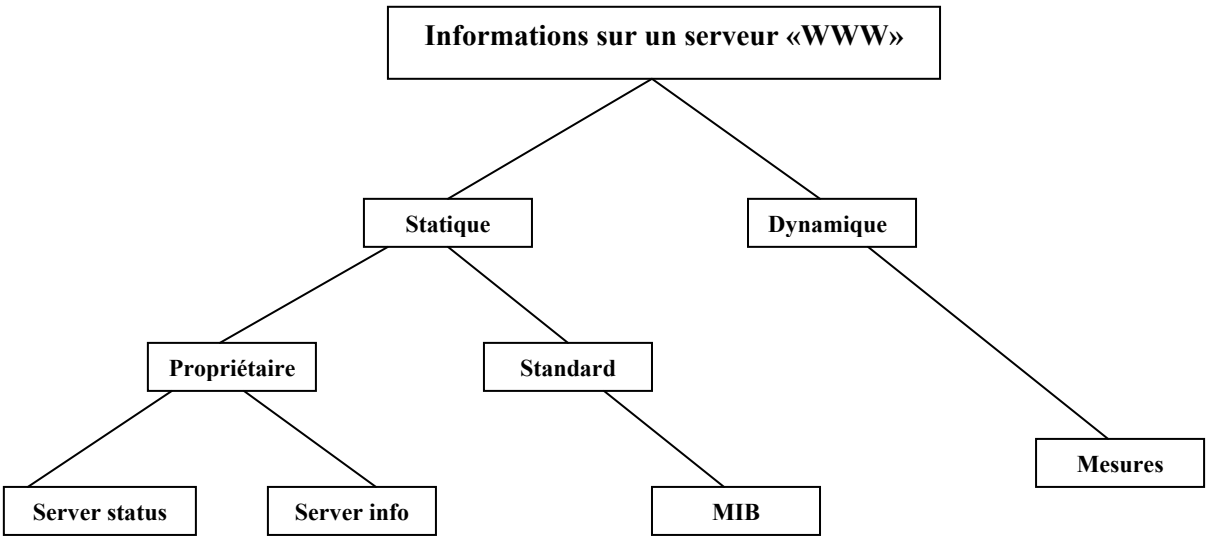


Figure 4-2 Informations sur un serveur «WWW»

La façon la plus classique d'utiliser les informations d'un serveur « www » c'est d'utiliser la MIB (Management Information Base). La [Réf 2594] nous donne une description exacte de la manière d'utiliser la MIB pour les services en ligne « www ». La plus part du temps les informations statistiques propriétaires représentent des caractéristiques du serveur

(current time, restart time, server uptime, le nombre des accès, l'usage CPU[22] et les performances) [Réf Apache]. Elles ne sont accessibles que par l'administrateur du serveur. Celui-ci peut aussi accéder au « server status report » c'est-à-dire (http://servername/server-status) et au « server configuration status reports » c'est-à-dire (http://servername/server-info). Or dans notre cas c'est l'utilisateur final qui a besoin d'accéder à différentes informations statistiques qui représentent en fait des paramètres de la qualité de service.

La [Réf 2594] a pour but de définir un ensemble de paramètres standard pour le gestion du services World Wide Web. « *The goal of this MIB is to define a standardized set of objects which lead to integrated and improved performance and fault management in a heterogeneous environment of WWW services. This MIB focuses on the service-oriented view ».* En s'appuyant sur cette référence on peut construire notre propre « MIB » accessible par l'utilisateur final.

Les paramètres définis dans la [Réf2594] sont séparés en 3 groupes principaux (Figure 4-3[23]):

- service ;
- protocole ;
- document.

[22] CPU : Central Processing Unit
[23] Nous avons gardé les mêmes noms des paramètres que dans la Réf.2594.

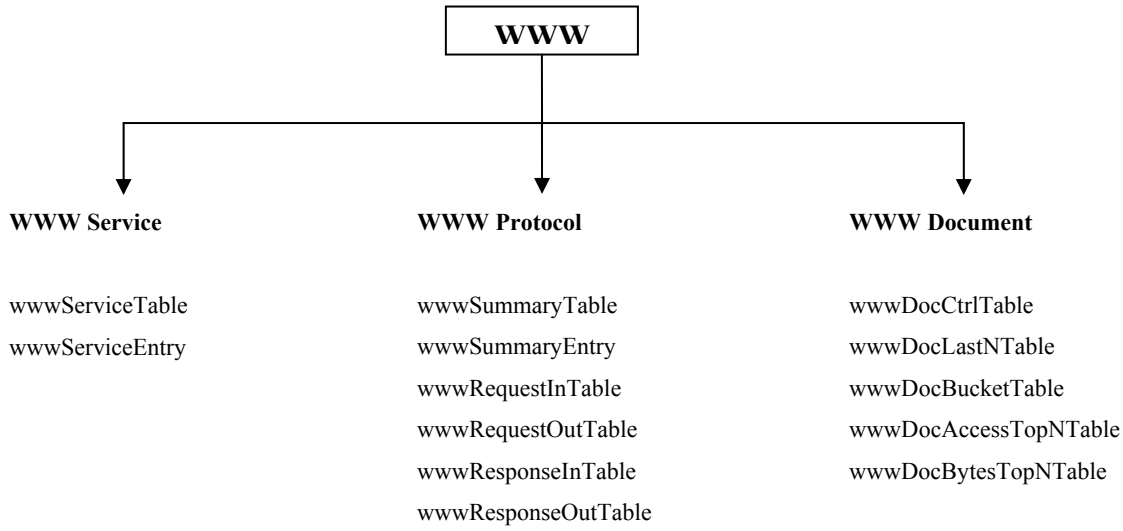


Figure 4-3 WWW Management Information Base

Le premier groupe contient les paramètres concernant tous les services WWW dirigés par un agent SNMP[24]. Cet agent fournit des informations concernant tout élément d'un réseau ou d'une machine autour d'un seul host. Une description de chaque paramètre est présentée dans le Tableau 4-5.

Paramètre	**Description**
wwwServiceTable	Le tableau des services WWW reconnus par l'agent SMTP
wwwServiceEntry	Contient plusieurs paramètres qui seront présentés dans le tableau 4.8

Tableau 4-5 WWW Service.

Le group « WWW Protocol » contient des paramètres liés au trafic, générés par les services WWW. Une description de chaque paramètre est présentée dans le Tableau 4-6.

Paramètre	**Description**

[24] SNMP : Simple Network Management Protocol

wwwSummaryTable	Le tableau des statistiques concernant les services WWW sur un host.
wwwSummaryEntry	Le tableau des statistiques concernant un service WWW particulier sur un host.
wwwRequestInTable	Le tableau des statistiques concernant les requêtes reçues concernant les services WWW sur un host.
wwwRequestInTable	Le tableau des statistiques concernant les requêtes générées concernant les services WWW sur un host.
wwwResponseInTable	Le tableau des statistiques concernant les réponses reçues concernant les services WWW sur un host.
wwwResponseOutTable	Le tableau des statistiques concernant les réponses générées concernant les services WWW sur un host.

Tableau 4-6 WWW Protocol

Le dernier group « WWW Document » contient des informations sur les documents qui ont été accédés sur un host. Une description de chaque paramètre est présentée dans le Tableau 4-7.

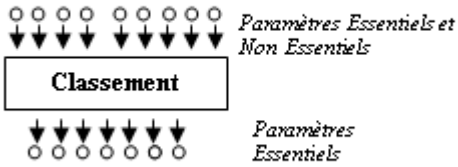
Paramètre	**Description**
wwwDocCtrlTable	Le tableau qui contrôle comment l'implémentation du MIB collecte et maintient les statistiques sur des différents documents.
wwwDocLastNTable	Le tableau qui contient les N derniers essais d'accès au service.
wwwDocBucketTable	Le tableau qui contient des informations pour l'administrateur en ce qui concerne les « buckets[25] » attribués à chaque service WWW.
wwwDocAccessTopNTable	Le tableau qui contient les noms des documents les plus accédés pour un « bucket » donné.
wwwDocBytesTopNTable	Le tableau qui contient les noms des documents qui ont généré plus de trafic dans un « bucket » donné.

Tableau 4-7 WWW Document

En tenant compte de la [Réf 2594] nous avons identifié les paramètres essentiels et non essentiels en ce qui concerne l'évaluation de la qualité de service de type WWW.

[25] Bucket : Mécanisme de contrôle d'accès, permettant de limiter les accès d'un utilisateur à certaines ressources, en ignorant simplement certaines de ses demandes en cas de congestion

4.2.2 Etape 2: Le classement



Au cours de cette étape les paramètres essentiels doivent être identifiés dans l'ensemble de paramètres trouvés au cours de la première étape. Les paramètres les plus importants quand on parle des services en ligne (de type www) sont la disponibilité et les performances de ces services. Selon ma propre expérience, étant un grand utilisateur de service en ligne depuis quelques années, les problèmes les plus courants que j'ai rencontrés, par ordre décroissant de sévérité sont :

- une réponse lente du serveur ou de la machine ;
- la connexion perdue avec le serveur ;
- l'impossibilité de trouver un serveur ;
- serveur atteint, mais l'accès aux informations impossible ;
- les URL perdus.

Tous ces problèmes ne tiennent pas compte de la bande passante et du type d'accès à l'Internet.

On considère donc la disponibilité et les performances des services en ligne des paramètres essentiels de l'évaluation de la qualité de service, pour des services en ligne www. On retrouve ces paramètres à travers les paramètres de la MIB présentés dans Tableau 4-5, Tableau 4-6, Tableau 4-7 et plus particulièrement le groupe de paramètres *wwwServiceEntry* et *wwwSummaryEntry*. Les 2 groupes seront présentés par la suite sous forme de tableau.

Le *wwwServiceEntry* contient des paramètres sur un service www particulier qui sont présentés dans le Tableau 4-8.

Paramètre	Description
wwwServiceIndex	Un nombre entier attribué à chaque type de service.
wwwServiceDescription	Une description textuelle du service.
wwwServiceContact	Le nom de la personne à contacter en ce qui concerne un service et aussi la moyenne de contacter la personne ex. webmaster@domaine.nom.
wwwServiceProtocol	Une identification du premier protocole à utiliser par le service (protocole DTP).
wwwServiceName	Le nom entier et le domaine auquel le service appartient.
wwwServiceType	Le type d'application nécessaire pour accéder au service.
wwwServiceStartTime	La date et l'heure de début d'accès au service.
wwwServiceOpenStatus	Indique le niveau opérationnel du service (en cours, hors service, arrêté, bloqué, redémarré)
wwwServiceLastChange	La date et l'heure précise du dernier changement du niveau opérationnel du service.

Tableau 4-8 Description wwwServiceEntry

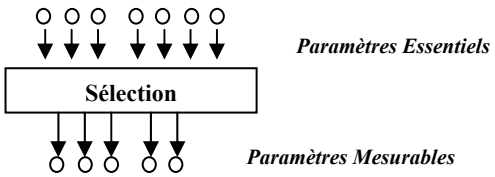
Le *wwwSummaryEntry* contient des paramètres sur un service www particulier, mais plus orienté protocole (dont réseau). Ces paramètres sont présentés dans le Tableau 4-9.

Paramètre	Description
wwwSummaryInRequests	Le nombre de requêtes reçues avec succès.
wwwSummaryOutRequests	Le nombre des requêtes générées.
wwwSummaryInResponses	Le nombre de réponses reçues avec succès.
wwwSummaryOutResponses	Le nombre de réponses générées.
wwwSummaryInBytes	Le nombre de bytes reçus.
wwwSummaryInLowBytes	Le plus petit nombre de paquets de 32 bytes sur le wwwSummaryInBytes.
wwwSummaryOutBytes	Le nombre de bytes envoyés.
wwwSummaryOutLowBytes	Le plus petit nombre de paquets de 32 bytes sur le wwwSummaryOutBytes.

Tableau 4-9 Description wwwSummaryEntry

Sur les 3 principaux groupes de paramètres de la MIB (Service, Protocole et Document) le dernier nous semble trop technique et donc nous l'avons enlevé. Dans les groupes Service et Protocole nous n'avons que les premiers éléments qui regroupent des paramètres essentiels en ce qui concerne l'évaluation de la qualité de service par l'utilisateur final. Les autres paramètres par leur technicité sont plus tournés vers les administrateurs réseaux des opérateurs ou fournisseurs de services.

4.2.3 Etape 3: Choix de mesures



Dans cette étape un choix doit être fait sur les paramètres essentiels qui peuvent être mesurés d'une manière automatique et économique. Pour cette raison faire des mesures sur un URL qui existe, mais qui n'a pas forcement de contenu n'est pas tout intéressant, ni pour l'utilisateur final, ni pour le fournisseur d'accès. Partant du ce principe, on constate que les 6 premiers paramètres du Tableau 4-8, ainsi que les 2 derniers ne sont pas adaptés à notre démarche.

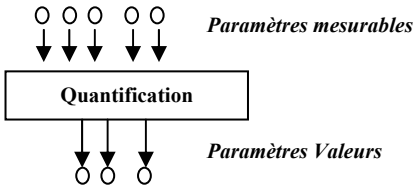
Comme nous l'avons déjà précisé dans le paragraphe précédent les paramètres de la qualité que nous testons pour les services en ligne sont la disponibilité et les performances du service. Le premier critère peut se traduire en termes de paramètre mesurable par *le temps d'accès* à un URL. Pour le deuxième (les performances), on considère les paramètres mesurables suivants : *le temps de transfert, la vitesse de transfert et la quantité transférée*. Ces paramètres mesurables se trouvent parmi les paramètres MIB suivants:

Paramètres de la qualité de service	**Paramètres mesurables de la qualité de service**	**Paramètres MIB**
Disponibilité	*le temps d'accès*	wwwSummaryInRequests
		wwwSummaryOutResponses
Performance	*le temps de transfert,*	wwwServiceStartTime
	la vitesse de transfert	wwwSummaryOutBytes
	la quantité transférée	

Tableau 4-10 Choix de paramètres MIB par rapport aux paramètres QoS

Après ce pas nous avons choisi les paramètres MIB mesurables.

4.2.4 Etape 4: Quantification des Paramètres mesurables



Dans cette étape, une valeur numérique doit être attribuée pour chaque paramètre mesurable. Cette valeur numérique peut être considérée comme une valeur seuil qui nous servira pour faire l'évaluation finale dans la dernière étape. Ces valeurs doivent être présentes dans les contrats de service (SLAs) signés entre les utilisateurs et les fournisseurs de services/les opérateurs. Dans notre cas nous sommes dans l'incapacité de donner des valeurs numériques parce que d'une part il n'existe pas de contrat signé avec un fournisseur de services et d'autre part si on prend le paramètre de *la quantité transférée* celle-ci dépend de chaque utilisateur. Tout de même, en se basant sur l'expérience de l'auteur le *temps*

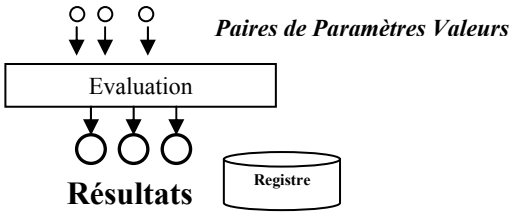
d'accès à un URL ne doit pas dépasser 15sec et la vitesse de transfert ne peut pas dépasser la vitesse actuelle proposée sur les réseaux GSM 9.6 kbit/sec. Dans le Tableau 4-11 nous avons un récapitulatif des paramètres valeurs pour les services en ligneWWW.

Paramètre	**Valeur**
le temps d'accès	**15 secondes**
le temps de transfert	**X secondes**
la vitesse de transfert	**< 9.6 kbit/sec**
la quantité transférée	**X kbit**

Tableau 4-11 Paramètres valeurs pour les services en ligne WWW

Dans notre tableau la valeur « X » veut dire un nombre entier qui n'est pas fixe et qui dépend de l'utilisation du web par chaque personne.

4.2.5 Etape 5: L'évaluation de la qualité de service



Cette dernière étape nous permet d'avoir des résultats concrets en ce que concerne l'évaluation de la qualité de service pour les services en ligne. Mesurer les paramètres de la qualité signifie dans ce cas comparer ces paramètres avec ceux que nous avons définis dans le Tableau 4-11.

A travers cette comparaison nous allons trouver des résultats qui prouvent que :

- l'utilisateur final a été pris en compte pour l'évaluation de la qualité de service ;
- les paramètres pris en compte sont directement observables et mesurables au point d'accès par l'utilisateur final ;

- ils sont reproductibles et comparables.

Une implémentation de cette évaluation de la qualité de service à l'aide des Agents Intelligents sera présentée dans le paragraphe 5.3.

4.3 Evaluation de la Qualité de Service pour le streaming vidéo

4.3.1 Le streaming vidéo

Dans le monde Internet d'aujourd'hui la vidéo est un moyen efficace de communication. Malheureusement, le volume important de données nécessaires pour constituer un fichier vidéo limite fortement son utilisation sur Internet. Les solutions de streaming permettent d'acheminer des données vers un utilisateur, qui peut commencer à les consulter avant qu'elles ne lui soient complètement parvenues et sans avoir besoin de les stocker localement. Il permet ainsi la diffusion d'une vidéo sous forme de **flux continu** sans téléchargement préalable de fichier, et donc sans délai d'attente pour le visiteur du site web (nous nous intéressons ici au cas de la vidéo, le principe étant identique pour des contenus audio).

Du côté de l'utilisateur, la visualisation des contenus fait appel à un **lecteur multimédia** et peut avoir lieu soit dans la page web consultée, soit dans le lecteur externe qui s'ouvre alors automatiquement. Ce logiciel décode le flux en temps réel et utilise une mémoire tampon (*buffer*) pour obtenir un débit régulier. Dans notre cas, l'utilisateur possède un téléphone mobile avec lequel il a la possibilité de lire des fichiers vidéo. Pour simuler ceux-ci nous avons utilisé le même outil que pour l'Agent web et le Mail, le J2ME Wireless Toolkit (la simulation revient à un accès de type HTTP à un serveur web).

Du côté du fournisseur de contenus, l'**encodage** peut se faire à partir de sources analogiques ou numériques (sortie directe d'un caméscope, cassette VHS, DVD, fichier vidéo...). Si le fichier encodé est destiné à être installé sur un serveur web, une technique couramment utilisée est de produire plusieurs versions en fonction de la possibilité d'accès de l'utilisateur (modem 56 K, liaison haut débit ou accès par le téléphone mobile). Bien entendu, plus la vitesse de connexion est grande, meilleures seront

la qualité de l'image et celle du son. Dans le cas d'une diffusion vidéo destinée à un large public, il est préférable d'utiliser un serveur spécifique – utilisant le protocole *Real Time Protocol* (RTP) [Réf 1889] (*e.g.* Real Media [Réf Real Media] ou Windows Media [Réf Windows Media]) adapté à la transmission de données en continu. Cela permet d'assurer une meilleure fluidité et de supporter de nombreux internautes connectés simultanément (rappelons qu'un serveur HTTP est prévu pour transférer de l'information par paquets et non par flux).

La vidéo sur Internet existe depuis longtemps, mais la qualité des lignes et la médiocrité des codecs de compression bas-débit ont empêché une large utilisation. De plus, Internet, ne permet pas de garantir un débit constant entre deux points (surtout s'ils sont éloignés). Le protocole de transfert des données sur Internet est TCP (Transfer Control Procotol Réf 1180TCP/IP Tutorial), mais comme le nom l'indique il contrôle le flux et donc il ne permet pas le temps réel. Pour le streaming, on utilise donc le protocole UDP (User Datagram Protocol [Réf. 768]) pour le transport. Pour les données, on utilise le protocole RTP (Realtime Transport Protocol [Réf. 1889]) qui utilise UDP. RTP inclut RTSP (Realtime Transport Streaming Protocol [Réf. 2326]) qui lui indique la qualité de ce qui est reçu.

En s'appuyant sur ces protocoles de transmission bas-niveaux, de nouvelles solutions technologiques permettent aujourd'hui l'accès au contenu audio-vidéo avec une qualité acceptable, en temps réel, en utilisant le réseaux Internet. Parmi ces solutions nous énumérons le progres réalisé pour assurer la bande passante (avec l'arrivée de l'ADSL [ref_ADSL] le débit est au maximum 14 fois supérieur au débit d'un modem 56K) et les techniques de compression de données (à l'aide de codecs standardisés [REF MPEG-4, REF_ITU-26L] ou propriétaires [REF_WindowsMedia, REF_RealNetworks]). Les nouvelles technologies de compression liées au streaming permettent une adaptation de la qualité de codage (donc du volume nécessaire pour représenter le contenu) en fonction du débit effectif. Le terme anglais pour cela est "**bandwidth negociation**".

Les enjeux technologiques et économiques liés à la diffusion de contenu multimédia sur Internet, ont conduit la communauté scientifique internationale à la mise en œuvre d'un nouveau standard "Multimedia Framework", mieux connu sous le nom MPEG-21 [Réf. MPEG 21]. MPEG-21 est une norme multimédia encore en élaboration, initiée par le

groupe MPEG (Moving Picture Experts Group). Elle est conçue pour « permettre une utilisation transparente et enrichie de ressources multimédia dans une vaste gamme de réseaux et d'appareils exploités par différents groupes ». Elle permet l'intégration des processus requis pour créer, manipuler, utiliser, gérer et diffuser des fichiers multimédias. La norme *MPEG-21* comprend des éléments qui appuient l'identification et la description de ressources numériques, le traitement et l'utilisation du contenu, la gestion et la protection de la propriété intellectuelle, etc.

4.3.2 Techniques de compression video

Selon la restitution totale ou la restitution partiale, mais pertinente, de l'information, il existe deux types de compression des données,:

4.3.2.1 Compression sans perte

Le fichier décompressé est identique au fichier original. Cette technique est largement utilisée pour les fichiers de données et les exécutables. Le taux moyen de compression typique n'est que 2:1, et est en tout état de cause nettement insuffisant pour exploiter des fichiers vidéo.

4.3.2.2 Compression avec perte

Avec ce type de compression, on atteint des taux de compression très élevé (300:1) mais au détriment de la qualité de l'image, qui pert de l'information mais sans que cela ne soit nécessairement détectable visuellement.

Dans le cas de la vidéo, on distingue par ailleurs, la compression temporelle et la compression spatiale.

4.3.2.3 Compression temporelle

Son objet est de ne stocker que ce qui est modifié lors du passage d'une image à une autre dans une séquence vidéo. Les images ainsi compressées peuvent être de deux types : *image clé* ou *image delta*. Les *images clés* sont des images de références, qui contiennent en elle-même, toute l'information. Les *images delta* ne contiennent que les pixels modifiés vis à vis de l'image précédente, qui peut être elle-même une *image clé* ou une *image delta*. La première image est nécessairement une *image clé*. Des techniques particulières, dont la **compensation de mouvement**, permettent d'optimiser la génération et la compression des *images delta*.

4.3.2.4 Compression spatiale

Cette compression s'applique exclusivement à une image donnée (clé ou delta), sans tenir compte des images environnantes. Il y a là différentes techniques : null suppression, RLE (Run Length Encoding), technique JPEG (Join Pictures Expert Group) utilisant la DCT (Discrete Cosine Transform).

Dans la compression temporelle MPEG spécifie 3 types d'images :

Les images I (Intra)

Ce sont des images codées sans aucune référence à d'autres images ; c'est à dire qu'on ne tient compte que du contenu de l'image elle-même et non des images environnantes. Elles autorisent l'accès aléatoire, soit l'accès à la séquence vidéo depuis n'importe lequel de ses points. Elles ont un taux de compression modéré, typiquement 2 bits par pixel codé.

Les images P (Prédites)

Ce sont des images codées par rapport à l'image de type I ou P précédente, grâce aux techniques de prédiction avec compensation du mouvement. Celles-ci n'étant pas parfaites, les erreurs se propagent puisqu'elles réutilisent les informations de l'image précédente. Leur taux de compression est nettement plus important que celui des images I.

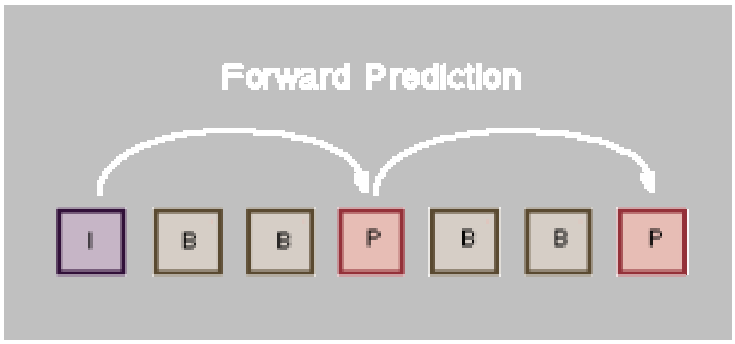


Figure 4-4 Relation entre les images de type I et P

Les images B (Bidirectionnelles)

Ce sont des images codées par interpolation entre les 2 types d'images I ou P précédente et suivante qui les encadrent. Ceci a le grand avantage d'offrir le meilleur taux de compression, sans pour autant propager les erreurs puisqu'elles ne sont jamais utilisées comme références. C'est ce type d'images qui offrent le taux de compression le plus fort.

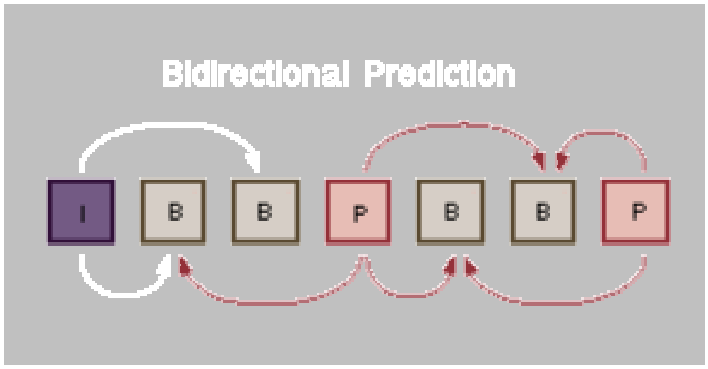


Figure 4-5 Relation entre les images de type I, P et B

Concernant ces 2 types d'images, il faut conserver davantage de données pour décrire des images de type I, d'où leur taille « maximale », alors que la taille des images de type P peut représenter moins d'un dixième de cette taille. Les images de type B sont les moins volumineuses. Les images P et B étant calculées à partir de l'image I, il est impossible de gérer conjointement une seule image I avec d'autres images, toutes de type P et B. Il est

nécessaire d'intercaler plusieurs images I, faute de quoi les erreurs se multiplient au détriment de la qualité de l'image.

4.3.3 Compression audio-video MPEG-1

Le travail du groupe MPEG s'est concrétisé pour la première fois en 1992 par la norme ISO/IEC 11172 [REF_ISO], plus largement connue sous le nom MPEG-1. Le but principal était de permettre le stockage et la reproduction vidéo sur support CD-ROM. Le débit envisagé par les spécifications était de l'ordre de 1,5 Mb/s dont 1,15 Mb/ pour la partie vidéo et de l'ordre de 350 Kb/s pour la partie son (stéréo) et données auxiliaires.

La compression MPEG-1 reprend une partie des spécifications de la norme JPEG [Réf. JPEG ISO/IEC 10918) concernant les techniques de codage d'images fixes et lui adjoint des traitements propres pour les séquences d'images. Par rapport à JPEG, elle procède à un deuxième type de compression afin de réduire davantage l'encombrement de toutes les données. Comme entre les images successives d'une séquence vidéo il existe une forte corrélation temporaire, MPEG exploite cette redondance en codant les différences entre les images.

Un flux de données MPEG-1 est constitué de deux couches :

- la couche système, qui stocke l'information temporelle nécessaire à la synchronisation vidéo et audio.
- la couche compression, qui inclue les flux audio et vidéo.

Les spécifications de la norme MPEG-1 sont publiées en trois parties distinctes [Réf Hervé]:

- MPEG system [Réf ISO/IEC 11172-1] définit la structure du multiplex MPEG-1 ;
- MPEG video [Réf ISO/IEC 11172-2] définit le codage vidéo MPEG-1 et
- MPEG audio [Réf ISO/IEC 11172-3] définit le codage audio MPEG-1.

La première partie définit les règles pour la constitution d'une « couche système » (***system layer***) regroupant audio, vidéo et données privées en un seul train, ainsi que les contraintes sur les trains élémentaires permettant d'assurer cette combinaison. Les cinq fonctions de base de la couche système sont les suivantes :

- *synchronisation de trains élémentaires compressés multiples à la reproduction ;*
- *combinaison de ces trains multiples en un seul train de données ;*
- *initialisation des mémoires tampons (**buffers**) au début de la reproduction ;*
- *gestion continue de ces buffers ;*
- *identification du temps par des marqueurs (**time stamps**).*

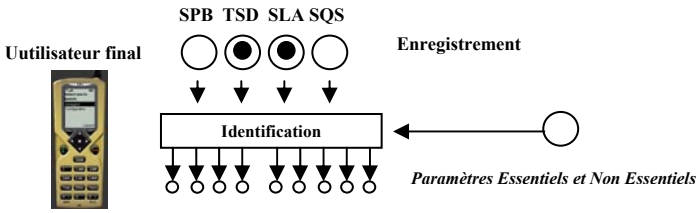
Chaque train élémentaire est découpé en paquets qui constituent un ***Packetized Elementary Stream*** (PES). Chaque paquet est constitué d'une en-tête de paquet (*packet header*) suivie des données proprement dites. L'en-tête du paquet commence par un code de départ sur 32 bits dont les 8 derniers identifient le type (audio, vidéo ou données privées) et le numéro du train élémentaire auquel le paquet de données appartient. Les paquets sont regroupés en *packs*, dont l'en-tête (pack header) fournit des informations de timing et de référence d'horloge système (System Clock Reference) sur 33 bits. Le premier pack d'un train MPEG-1 contient obligatoirement une en-tête système (***system header***). C'est un paquet spécial qui résume l'ensemble des paramètres système utilisés au cours de ce train (débit maximum, identifications des trains élémentaires audio, vidéo et données, taille minimale du buffer…). Ce paquet servira dans nos tests pour identifier les paramètres audio et vidéo nécessaire pour l'évaluation de la qualité de service.

Les tests ont été réalisés sur des fichiers audio/vidéo de type MPEG-1, principalement à cause de la large utilisation de ce format pour représenter le contenu audio-vidéo sur Internet. Les fichiers de type MPEG-2 sont trop importants en termes de taille pour être visualisé sur des téléphones mobiles. Même si les spécifications de la norme MPEG-4 sont déjà publiées, le manque du contenu conforment au standard n'a pas permis la réalisation

des tests pour ce format. Toutefois, nos procédures de test sont génériques et facilement adaptables pour le prendre en compte dans une future investigation.

Cette introduction concernant les fichiers MPEG et l'accès par le flux vidéo nous permettront par la suite de comprendre le choix de nos paramètres concernant l'évaluation de la qualité de service.

4.3.4 Identification de tous les paramètres de la qualité



Dans ce premier pas, tous les paramètres essentiels et non essentiels de la qualité de service pour le streaming vidéo doivent être identifiés. Comme on l'a vu, les serveurs de streaming doivent savoir gérer l'adaptation et l'optimisation du flux et du contenu. L'adaptation au réseau et au terminal doit se faire en temps réel. Des réseaux de distribution de contenus mobiles se développeront, sur le modèle des « Content Delivery Networks » de l'Internet [Réf. Akamai]. Aux extrémités des réseaux, au plus près des utilisateurs, des serveurs multimédia gèreront une part de la distribution et de l'adaptation au contexte utilisateur. Le contenu doit être adapté pour maximiser l'expérience utilisateur. Donc en terme de qualité de service du point de vue de l'utilisateur final pour le streaming vidéo il est important de prendre en compte la qualité du contenu fourni (image, son), la qualité d'accès au contenu (vitesse d'accès, bande passante) ainsi que la facilite d'utilisation du contenu. Pour les paramètres concernant le contenu nous nous sommes appuyés sur la norme ISO 11172-1 [Réf. ISO 11172-1] et surtout sur la partie intitulée « **system header** » (voir la section 4.3.2). Ces paramètres (audio et vidéo) sont présentés dans les 2 tableaux suivants.

Caractéristique	Paramètres	Description
Audio	*Couche*	Il y a trois sous couches de compression audio dans les formats MPEG-1: Layer I, Layer II et Layer III. De Layer I à Layer III, les méthodes de codage et les performances augmentent. Les trois couches Layer I,II,III gardent une compatibilité ascendante.
	fréquence d'échantillonnage (kHz)	Le nombre d'échantillons du son prélève à chaque seconde.
	Bit rate_index	Le nombre de bits par seconde qui ont été utilisé pour coder la séquence audio
	Mode_extension	C'est des bits qui sont utilisés dans le mode stéréo et qui donne les 2 types de codage : intensity-stéréo et stéréo.
	Copyright	Le bit indique s'il existe un copyright ou pas sur la séquence audio
	Original/Home	Le bit indique l'originalité de la séquence audio
	Nombre de trames	Le nombre de trames audio qui correspond à la séquence vidéo visionnée

Tableau 4-12 Paramètres essentiels et non essentiels audio pour une séquence vidéo MPEG 1

Caractéristique	Paramètres	Description
Vidéo	*Taille de la trame (horizontale et verticale)*	C'est la taille mesurée en pixels.
	Format géométrique	Le format géométrique de l'image.
	Pel_aspect_ratio	Série de 4 bits pour définir le type d'image (VGA, 16:9, CCIR601).
	Picture_rate(nombre de trames par seconde)	Série de 4 bits qui correspond à un nombre bien définit de trames par secondes (entre 23.976 à 60).
	Bit rate_index	Le nombre de bits par seconde qui ont été utilisée pour coder la séquence vidéo.
	Nombre de trames I	I : Codage comme une image fixe, nécessaire car la séquence nécessite un commencement.
	Nombre de trame P	P : Image calculée à partir de l'image de type P ou I la plus récemment calculée (Chaque bloc d'une image peut être codé par des méthodes différentes).
	Nombre de trame B	B : Image codée à partir des 2 images de type I ou P les plus récentes l'une dans le passé, l'autre dans le futur.
	Nombre de trames	Le nombre de trames image qui correspond à la séquence vidéo visionnée.
	Vbv_buffer_size	C'est un numéro entier codé sur 10 bits qui définit la taille du VBV (Video Buffering Verifeer) qui est le buffer nécessaire pour décoder la séquence vidéo
	Constrained_parameter_flag	C'est un 'flag » codé sur 1 bit qui a la valeur 1 dans certaines cas [Réf ISO 11172-3]
	Load-intra-quantizer_matrix	Paramètre utilisé pour le codage des coefficients DCT (**Discret Cosinus Transform**) [Réf ISO 11172-3]
	Intra_quantizer_matrix	Paramètre utilisé pour le codage des coefficients DCT (**Discret Cosinus Transform**) [Réf ISO 11172-3]
	Load_non_intra_quantizer_matrix	Paramètre utilisé pour le codage des coefficients DCT (**Discret Cosinus Transform**) [Réf ISO 11172-3]
	non_intra_quantizer_matrix	[Réf ISO 11172-3]
	Extension_start_code	Suite de bits qui correspond à la suite 000001B5 en hexadécimale. Elle définit le début de l'extension des données.
	Sequence_extension_data	Réservée
	User_data_start_codde	Suite de bits qui correspond à la suite 000001B2 en hexadécimal. Elle définit le début des données.de l'utilisateur.
	User_data	Paramètre qui est définit par les utilisateurs pour des applications spécifiques.

Tableau 4-13 Paramètres essentiels et non essentiels image pour une séquence vidéo MPEG 1

Etant donné que l'accès au contenu est basé sur un accès Internet, les paramètres choisis sont les mêmes qu'au paragraphe antérieur (voir section **4.2.1).** Nous n'allons pas reprendre les arguments concernant les paramètres de la qualité de service retenus, mais

juste les paramètres essentiels et mesurables adaptés a un contenu audio/vidéo. Ces paramètres sont présentés dans le tableau suivant :

Paramètres de la qualité de service	**Paramètres mesurables de la qualité de service**
Disponibilité	Vitesse de transfert (kbit/sec)
	Nombre de trames transférées/sec

Tableau 4-14 Paramètres essentiels et mesurables pour l'accès à une séquence audio/vidéo

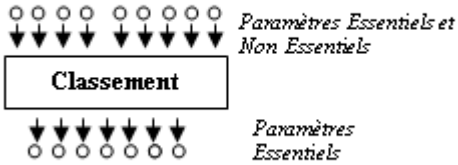
Pour compléter la liste de paramètres essentiels et non essentiels de la qualité de service pour le streaming vidéo, il est important de prendre en compte la façon dont l'utilisateur final procède pour accéder à une séquence audio/vidéo. Des paramètres comme la fiabilité ou la facilité d'utilisation (voir Tableau 4-1.) peuvent être adaptées à l'utilisation du streaming vidéo. Dans le tableau suivant nous avons justement présentés ces paramètres comme des paramètres essentiels et adaptés à notre besoin.

Paramètres de la qualité de service	**Sous paramètres**
Fiabilité	Maturité: Fréquence des défaillances (MTF), Densité des défauts, Stabilité du produit.
Facilité d'utilisation	Facilité de compréhension, Facilité d'opération, Facilité d'apprentissage, Qualité de la documentation

Tableau 4.15 Paramètres essentiels du contenu, d'une séquence audio/vidéo

Ayant comme principale base la norme ISO 11172-3, dans cette étape nous avons situé l'ensemble de paramètres essentiels et non essentiels pour l'évaluation de la qualité de service pour le streaming vidéo.

4.3.5 Etape 2: Le classement



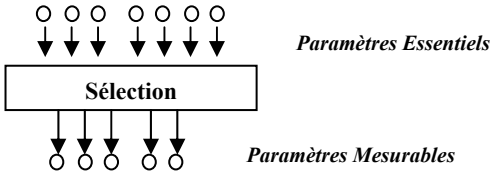
Au cours de cette étape les paramètres essentiels doivent être séparés de l'ensemble de paramètres trouvés au cours de la première étape**.** Pour obtenir ces paramètres, nous nous sommes appuyés sur notre propre expérience concernant les fichiers audio/vidéo. L'outil qui nous a inspiré est VirtualDub [Réf. VirtualDub] qui permet différentes opérations sur un fichier vidéo, une des options étant la possibilité d'obtenir les paramètres audio et vidéo exacts du fichier. Pour obtenir donc les paramètres essentiels de la qualité de service pour le streaming vidéo nous avons trié les paramètres vidéo et audio, tout en gardant les paramètres d'accès et de contenu. Ces paramètres essentiels sont présentés dans le tableau suivant :

Caractéristique	**Paramètres**
Audio	*Couche*
	Fréquence d'échantillonnage (kHz)
	Bit rate_index
	Nombre de trames
Vidéo	*Taille de la trame (horizontal et vertical)*
	Format géométrique
	Picture_rate(nombre de trames par seconde)
	Bit rate_index
	Nombre de trames I
	Nombre de trames P
	Nombre de trames B
	Nombre de trames
Disponibilité	*Vitesse de transfert (kbit/sec)*
	Nombre de trames transférées/sec
Fiabilité	*Maturité: Fréquence des défaillances (MTF), Densité des défauts, Stabilité du produit.*
Facilité d'utilisation	*Facilité de compréhension, Facilité d'opération, Facilité d'apprentissage, Qualité de la documentation*

Tableau 4-15 Paramètres essentiels de la qualité de service pour le streaming audio/vidéo

Après cette étape les paramètres essentiels et les sous paramètres essentiels pour l'évaluation de la QoS pour le streaming vidéo ont été déterminés.

4.3.6 Etape 3: Choix de mesures

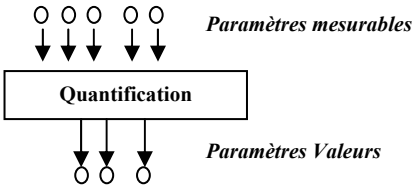


Dans cette étape un choix doit être fait sur les paramètres essentiels qui peuvent être mesurés d'une manière automatique et économique. Les paramètres essentiels que nous avons retenus sont les paramètres techniques audio et vidéo ainsi que les paramètres d'accès. Les autres paramètres sont des paramètres relatifs à chaque utilisateur final et de la façon dont il se sert du streaming vidéo. De plus à l'heure actuelle aucun opérateur de téléphonie mobile n'offre ce type de services donc il est quasiment impossible de tester des paramètres de ce type. Les paramètres retenus sont présentés dans le prochain tableau.

Caractéristique	**Paramètres**
Audio	*Layer*
	Fréquence d'échantillonnage (kHz)
	Bit rate_index
	Nombre de trames
Vidéo	*Taille de la trame (horizontal et vertical)*
	Format géométrique
	Picture_rate(nombre de trames par seconde)
	Bit rate_index
	Nombre de trames I
	Nombre de trame P
	Nombre de trame B
	Nombre de trames
Disponibilité	*Vitesse de transfert (kbit/sec)*
	Nombre de trames transférées/sec

Tableau 4-16 Paramètres essentiels mesurables de la qualité de service pour le streaming audio/vidéo

4.3.7 Etape 4: Quantification des Paramètres mesurables



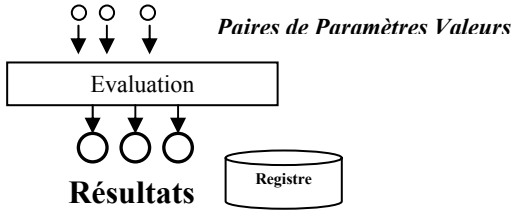
Dans cette étape une valeur numérique doit être attribuée pour chaque paramètre mesurable. Cette valeur numérique peut être considérée comme une valeur seuil qui nous servira à faire l'évaluation finale dans la dernière étape. Ces valeurs doivent être présentes dans les contrats de service (SLAs) signés entre les utilisateurs et les fournisseur de service/les opérateurs. Pour trouver ces valeurs nous nous sommes appuyés sur la norme ISO 11173.

Caractéristique vidéo	**Paramètre**	**Valeur**
Audio	*Type*	**Layer I, Layer II, Layer III**
	Nombre de trames	**X**
	Sampling rate (kHz)	**44.1, 48 ou 32**
	Bit rate (kbit/sec)	**32-448**
Image	*Taille de la trame*	**X pixels x Y pixels< 4096x4096**
	Format géométrique	**SQUARE**
	Nombre de trames/sec	**23.976,24, 25, 29.97, 50, 59.94, 60**
	Bit rate (kbit/sec)	**64 kbit/sec à 1,5 Mbit/sec**
	Nombre de I	**X**
	Nombre de P	**X**
	Nombre de B	**X**
	Nombre de trames	**X**
Performance réseaux	*Vitesse de transfert (kbit/sec)*	**X**
	Nombre de trames transférées/sec	**X**

Tableau 4-17 Paramètres valeurs pour le service de streaming audio/vidéo

Dans notre tableau la valeur « X » veut dire un nombre entier qui n'est pas fixe et qui dépend de l'utilisation du streaming vidéo de chaque personne.

4.3.8 Etape 5: L'évaluation de la qualité de service



Cette dernière étape nous permet d'avoir des résultats concrets en ce que concerne l'évaluation de la qualité de service pour les services on-line. Mesurer les paramètres de la qualité signifie dans ce cas comparer ces paramètres avec ceux que nous avons définis dans le Tableau 4-17.

A travers cette comparaison nous allons trouver des résultats qui prouve que :

- l'utilisateur final a été pris en compte pour l'évaluation de la qualité de service ;
- les paramètres pris en compte sont directement observables et mesurables au point d'accès par l'utilisateur final ;
- sont reproductibles et comparables.

4.4 Résumé du chapitre

Dans ce chapitre nous avons appliqué le modèle théorique de l'évaluation de la qualité de service proposé au chapitre 3 pour différents types de services : mail, accès de type www et streaming vidéo. Nous avons obtenu une série de paramètres essentiels et mesurables dont les valeurs de seuils ont pu être déterminées pour la plupart. Afin d'évaluer la qualité de service, ces paramètres essentiels et mesurables obtenus après des tests réels seront comparés aux valeurs de seuils que nous avons déterminées. La plus grande difficulté dans cette démarche a été justement de trouver ces valeurs de seuils. L'absence des modèles de SLA (Service Level Agreement) ou même l'absence des services de streaming vidéo pour

des téléphones mobiles nous ont obligé à laisser certains paramètres sans valeur de seuil exacte.

Une implémentation de l'évaluation de la qualité de service pour le mail, l'accès de type www et le streaming vidéo à l'aide des Agents Intelligents est présentée dans le chapitre suivant.

5 Un prototype d'implémentation avec des Agents Intelligents pour simuler l'évaluation de la qualité de service

Le chapitre 5 décrit une implémentation de l'évaluation de la qualité de service en utilisant la technologie d'Agents Intelligents. Nous présentons une plateforme multiagents appelé ***Jade-LEAP*** *et un prototype d'agent d'évaluation de la QoS pour des services en ligne (www), le courrier électronique et le streaming vidéo. Ce chapitre vise à prouver que des agents peuvent servir à une évaluation automatisée de la qualité de service pour différentes applications et services avec une interface proche du téléphone mobile actuel. Ils réalisent les 5 étapes de notre procédure d'évaluation de QoS. L'utilisation des agents mène à des résultats qualitatifs et quantitatifs, reproductibles et surtout simples à mettre en oeuvre par les utilisateurs des services.*

5.1 Implémentation des Agents Intelligents pour mettre en oeuvre l'évaluation de la qualité de service pour différents services mobiles

La mise en œuvre de l'évaluation de la QoS a été réalisée par une simulation sur une plateforme utilisant la technologie des Agents Intelligents. Ce choix de réalisation a été fait étant donné l'absence d'une plateforme de test réelle accessible sur le marché actuel de télécommunications.

Pour exécuter les cinq étapes de l'évaluation de la QoS *(l'Identification, le Classement, la Sélection, la Quantification et l'Evaluation*) à l'aide de notre *Agent QoS* (voir la section 3.4), nous rappelons quelques précisions sur l'implémentation:

- les agents ont besoin d'un environnement de travail sur différentes machines (PC, SUN, etc.…) avec différents systèmes d'opération (Linux, Unix, Win 9x/2000/NT) ;
- un langage de programmation orienté-objet flexible et adaptable (notre choix est Java) ;
- un langage approprié de communication entre les agents (ACL dans notre cas) ;

L'utilisation des Agents Intelligents nous oblige à penser à l'évolution de la programmation orientée-objet vers la programmation orientée-agent. Cette programmation offre une façon plus « naturelle » [Réf Garn] de concevoir des systèmes et sous-systèmes

distribués sur différentes machines. Elle a été introduite par Yoav Shoham [Réf. Shoham] comme un nouveau « *paradigme de programmation basé sur une vue sociétale de la programmation* ». L'approche fondamentale de ce type de programmation est justement le développement de systèmes multi agents (SMA).

Notre implémentation sera faite à l'aide d'un système multi agents. L'environnement de développement de systèmes multi agents devrait contenir certaines caractéristiques importantes [Réf Garneau]:

- offrir un support simple pour le déploiement inter-machine ;
- offrir un support pour les bases de données ;
- faciliter te travail d'écriture du code ;
- permettre aux personnes moins expérimentées en programmation orientée-agent de créer facilement des SMA sans connaître tous les détails d'implémentation ;
- permettre aux développeurs plus expérimentés d'accéder et d'interagir directement avec les différents composants du système ;
- faciliter d'extension du code source ;
- décharger les développeurs de l'implémentation du système de communication et des protocoles utilisés pour transporter les messages ;
- posséder un utilitaire de « débuggage » (déverminage), une interface utilisateur facilitant le développement et un générateur automatique de code source ;
- être supporté par une documentation appropriée.

Plusieurs outils (éléments logiciels offrant des services pour le développement de SMA) de différents types ont été développés récemment pour la programmation orientée-agent. Parmi ces outils nous trouvons :

- **ZEUS** [Réf. Zeus]
- **JADE** [Réf. Jade]
- **AgentTool** [Réf. AgentTool]
- **AgentBuider** [Réf AgentBuilder]
- **MadKit** [Réf. Madkit].

Pour choisir un de ces outils nous nous sommes basés sur plusieurs critères :

- facilité de la programmation ;
- compatibilité avec le langage Java ;
- extensibilité du code (les utilitaires fournis par les outils comme les modules, les agents prédéfinis ou le code généré doivent être facilement modifiables et offrir la possibilité de rajouter du code à celui existant) ;
- possibilité d'implémentation avec des systèmes CDLC MIDP[26];
- possibilité de faire abstraction des mécanisme de communication, interaction entre les agents ;
- simplicité de l'installation et de l'utilisation de la plateforme ;
- documentation de qualité.

Après évaluation nous avons choisit l'outil **JADE**. Ce choix a été fait pour les raisons suivantes en rapport avec nos critères :

- plateforme assez facile à mettre en place

[26] Voir section 3.5.4

- possibilité d'extension de la plateforme JADE avec des bibliothèques LEAP (Réf.LEAP) qui permet l'implémentation et la simulation sur des systèmes CDLC MIDP ;

- existence d'exemples sur lesquels nous nous sommes appuyés pour construire notre propre code ;

- documentation claire et complète.

5.2 Plateforme JADE-LEAP

Dans un premier temps nous allons présenter la plateforme JADE séparément et ensuite l'extension avec les bibliothèques LEAP.

JADE (Java Agent DEvelopement Framework) est en logiciel gratuit créé par le laboratoire TILAB [Réf. TILAB]. JADE permet le développement de systèmes multi agents et d'applications conformes aux normes FIPA [Réf. FIPA] pour les Agents Intelligents. Ce logiciel contient 2 produits : une plateforme multi agents conforme FIPA et un outil pour le développement des agents JAVA.

L'outil possède trois modules principaux :

- ***DF « Director Facilitor »*** fournit un service de « pages jaunes[27] » à la plateforme ;

- ***ACC «Agent Communication Channel »*** gère la communication entre les agents ;

- ***AMS « Agent Management System »*** supervise l'enregistrement des agents, leur authentification, leur accès et l'utilisation du système.

Ces trois modules sont activés à chaque démarrage de la plateforme. (Figure 5-1).

[27] Le service de « pages jaunes » permet d'avoir le nom et l'adresse des agents présentent sur la plateforme

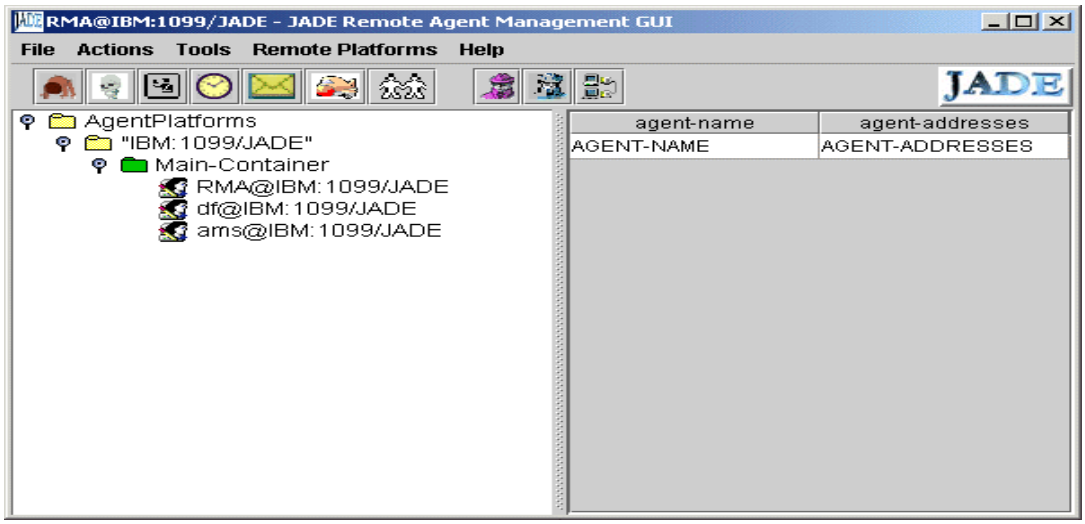


Figure 5-1 Lancement plateforme Jade

La structure de la celle-ci correspond au standard FIPA. Elle est représentée dans la figure suivante :

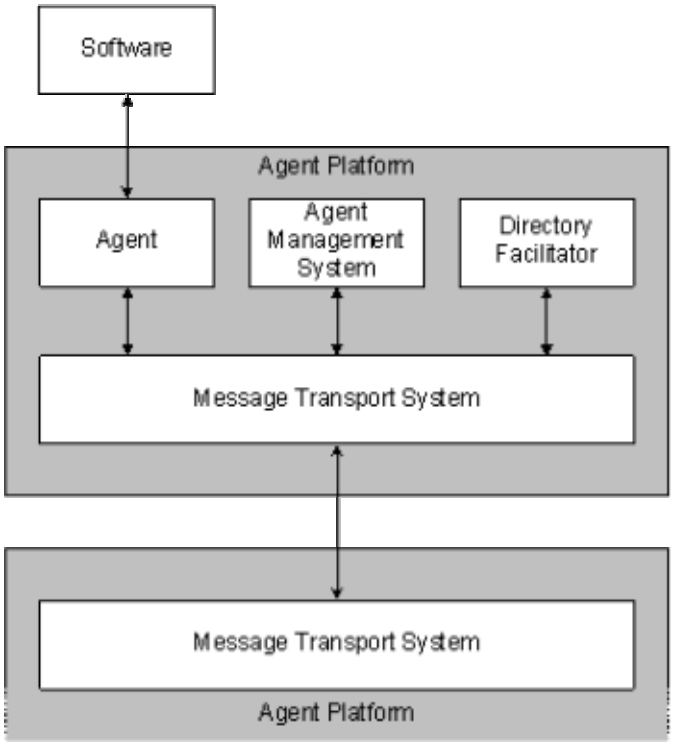


Figure 5-2 Architecture de référence pour les Plateformes FIPA

L'Agent Management System (AMS) est l'Agent principal de cette plateforme et correspond au document FIPA no 23 [Réf FIPA23]. Les autres agents peuvent se trouver sur des machines différentes et communiquent par le langage FIPA ACL (Agent Communication Language). Les Agents sont implémentés comme des « threads » Java et sont contenus dans des *« Agent Containers »* qui fournissent le support d'exécution de l'Agent. Sur la plateforme, il existe un container principal « *Main Container* » qui contient l'Agent AMS et les Agents existants sur la plateforme elle-même. Un container à part est crée à chaque utilisation de la plateforme par des Agents qui se trouve sur des hosts différents. Un éditeur est disponible pour l'enregistrement et la gestion des Agents. Cet éditeur propose des ontologies spécifiques FIPA.

Pour faciliter l'application et l'administration de la plateforme, jade dispose dune boite à outils « jade.tools » qui contient :

- *Le Remote Management Agent* (RMA) qui peut être une console graphique pour le contrôle de la plateforme ;
- *Le Dummy Agent* qui est un outil pour contrôler et trouver les erreurs dans les Agents ;
- *Le Sniffer Agent* est un outil qui peut intercepter et afficher des messages ACL entre les Agents ;
- *Le IntrospectorAgent* est l'outil qui contrôle la vie des Agents ;
- *Le* SocketProxyAgent est un Agent simple qui peut être considéré comme un lien bidirectionnel entre la plateforme et une autre connexion TCP/IP ;
- *Le DF GUI* est l'interface graphique par défaut utilisé par le DF (Directory Facilitor).

Aucune autre interface n'est disponible pour le développement et l'implémentation. Pour compléter cette lacune nous avons utilisé les bibliothèques **LEAP (Light Extensible Agent Plateform)** (voir Figure 5-3) qui ont été développées dans le cadre du projet

« LEAP» auquel ont participé différents opérateurs (British Telecom, TILAB) et constructeurs de mobiles comme Motorola, Broadcom Eireann Research et Siemens.

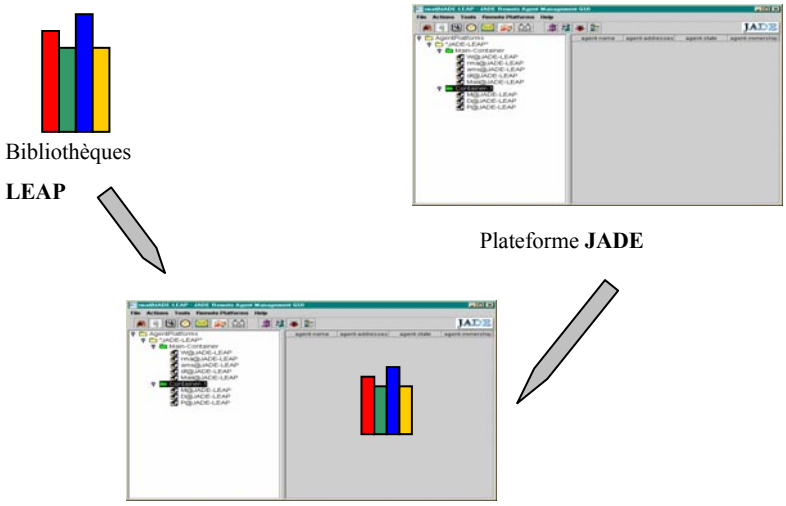


Figure 5-3 Plateforme **JADE-LEAP**

L'utilisation des bibliothèques LEAP remplace quelques parties du cœur de fonctionnalités de la plateforme JADE tout en offrant la possibilité d'implémenter et de faire des simulations sur des systèmes qui ont des limites au niveau mémoire et qui peuvent être compatible avec des environnements Java mobiles (CDLC MIDP). JADE-LEAP garde la même structure que JADE sauf qu'en plus des containers normaux le système offre la possibilité de construire des « *lightweight* » containers qui peuvent être exécutés sur des environnements Java MIDP. Les 2 types de container offrent le même ensemble d'API (Application Programming Interface) à l'utilisateur ce qui conduit à une large utilisation de la plateforme JADE-LEAP sur différents dispositif (du téléphone au PC), systèmes d'opérations ou pour différents types de réseaux (câbles ou sans fils). Ceci signifie qu'un agent qui a été conçu pour un PC peut très bien fonctionner sur un téléphone mobile avec un système MIDP embarqué sans changement du code. La figure suivante résume cette approche (Figure 5-4).

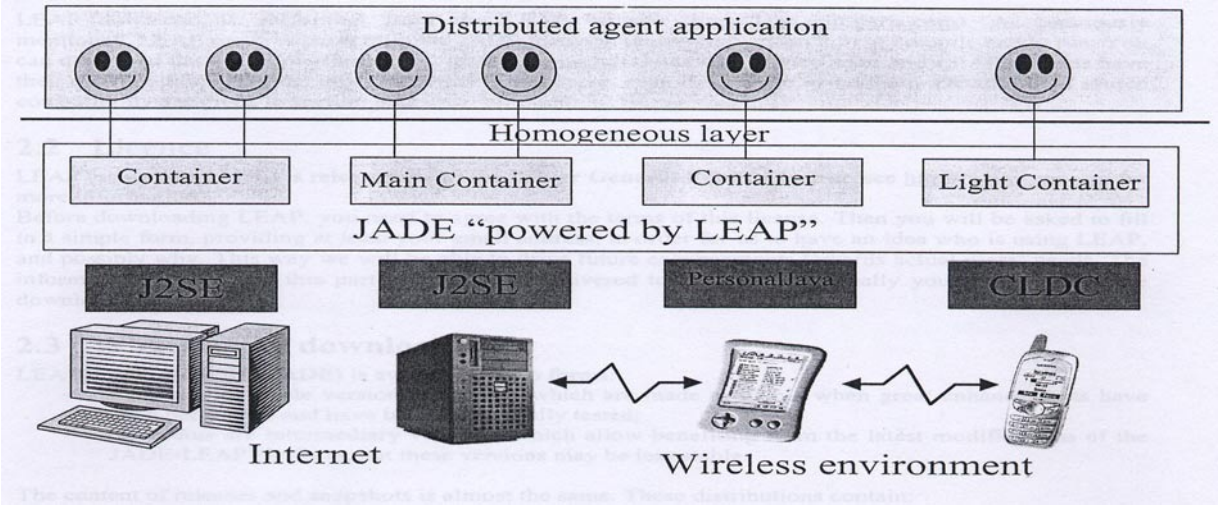


Figure 5-4 Couche fonctionnelle de la plateforme JADE-LEAP

Ce concept est intéressant pour nous dans la mesure où nous pouvons faire des simulations et des mesures à l'aide d'un outil qui nous donne une première approche des réseaux et des terminaux mobiles actuels.

Dans les prochains paragraphes nous allons présenter la conception et les simulations à l'aide d'agents intelligents qui ont été développées sur la plateforme JADE-LEAP pour l'évaluation de la qualité de service en ce qui concerne l'accès web, le mail et le streaming vidéo.

5.3 Architecture d'un Agent pour évaluation de la QoS pour les services on-line (web)

5.3.1 L'Agent Web

L'agent « Web » pour l'évaluation de la qualité de service pour les services on-line de type « www » a été construit en s'appuyant sur le notre modèle en 5 étapes de l'évaluation de la QoS.

Les principales caractéristiques que nous avons mesurées sont la disponibilité d'un serveur et la vitesse de transfert. Les couples paramètres valeurs sont représentes dans le tableau 5.1 (Tableau 5-1).

Paramètre	**Valeur**
temps d'accès	**15 secondes**
temps de transfert	**X secondes**
vitesse de transfert	**< 9.6 kbit/sec**
quantité transférée	**X kbit**

Tableau 5-1 Paramètres valeurs pour les services on-line WWW

La disponibilité du serveur a été traduite par le temps de d'accès au serveur. Dans notre agent nous avons aussi implémenté la possibilité de connaître la quantité d'information qui a été téléchargée au cours d'une connexion à un serveur. Ce paramètre est important si nous sommes dans la situation d'un réseau mobile à base de paquets (exemple le GPRS) où la facturation se fait par rapport au nombre de Koctets transférés. L'utilisateur final a non seulement une vision du site auquel il veut accéder, mais aussi il peut gérer la quantité d'informations qu'il a téléchargée.

Notre **Agent Web** est constitué en fait de 4 agents différents :

- un agent que nous avons appelé **Ping** qui mesure le temps de connexion à un serveur ;
- un agent que nous avons appelé **Download** qui mesure la vitesse de transfert et qui fait le téléchargement si c'est le cas ;
- un agent que nous avons appelé **Writer** qui a comme tâche d'écrire dans la base de données les informations qui lui ont été envoyées par l'Agent Manager ;
- un agent que nous avons appelé **Manager** et qui gère les 3 premiers agents.

Le principe de fonctionnement des 4 agents est présenté dans le schéma suivant :

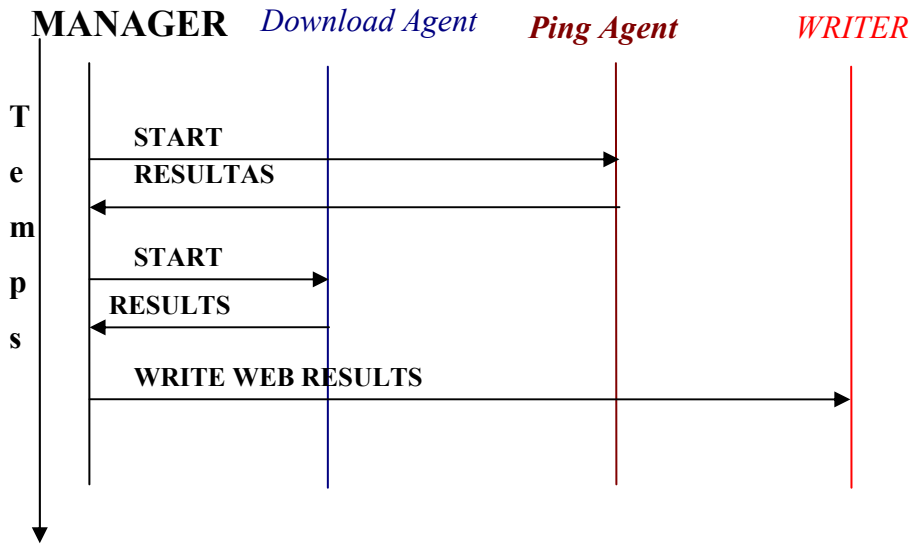


Figure 5-5 Description du fonctionnement de l'Agent Web »

L'Agent Manager possède une interface graphique qui lit les données sur le mobile et qui les envoies vers l'agent **Ping**. Celui-ci mesure le temps de connexion vers le serveur et il retourne comme résultats l'adresse du serveur, le nombre de Kbits qui ont été lus et le temps de connexion. Le **Manager** lance ensuite l'agent **Download** sur le même site. Il retourne comme résultats, la vitesse de transfert et fait le téléchargement si nécessaire[28]. Une fois les résultats arrivés, le **Manager** par l'intermédiaire de l'interface graphique présente une partie des résultats (pour des raisons de taille de l'écran du mobile) sur le mobile. L'intégralité des résultats est ensuite transmise à l'agent **Writer** qui les insère dans la base de données.

Les agents **Ping**, **Download** et **Manager** sont des agents qui se trouvent sur le mobile lui-même, tandis que l'agent **Writer** se trouve sur la plate-forme **Jade-LEAP**.

[28] L'utilisateur peut faire un test simple sur un serveur juste un allant sur un site (par exemple www.monsite.fr) ou il peut aussi télécharger un fichier d'un site précis (par exemple www.monsite.fr/fichier.extension)

A chaque agent correspond une ou plusieurs classes Java fonction de sa complexité. L'Agent Manager contient les classes suivantes (voir Figure 5-6): TestAgent (le corps de l'agent), TestGUI (la liaison entre l'agent et l'interface), TestGUIMIDP (interface input) et FormSubClass (interface output). Seul cet agent possède les classes qui gèrent l'interface graphique. Comme les agents s'exécutent sur des files d'exécution différentes (en anglais « thread »), l'interface graphique reste toujours active.

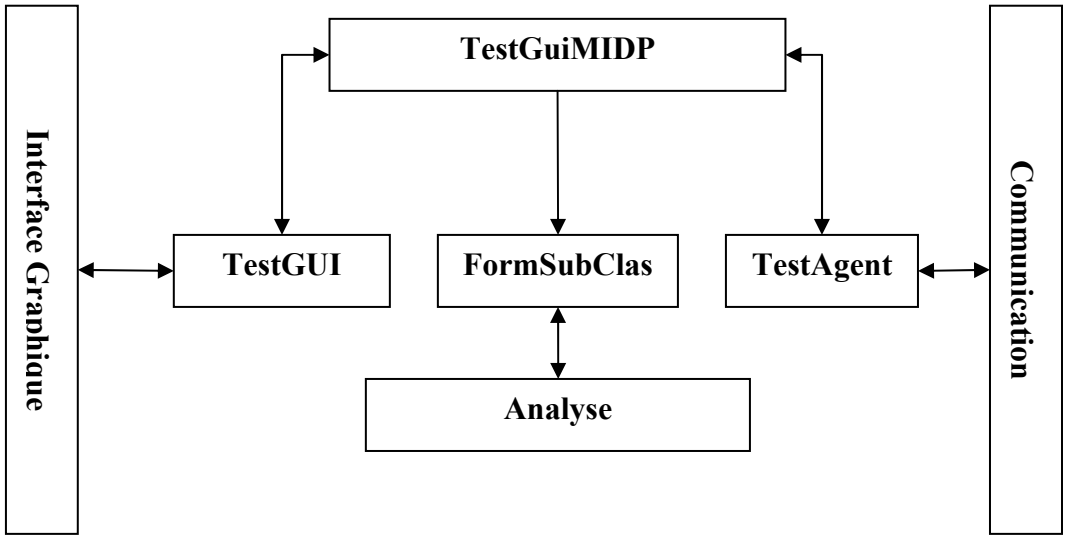


Figure 5-6 Structure interne de l'Agent Manager

Les agents Ping et Download ont comme classes pAg et dAg (voir Annexes) qui représentent le corps de chaque programme.

L'agent Writer sera présenté dans la dernière partie avec le streaming vidéo.

Dans le schéma Figure 5-7 nous avons résumé l'architecture de l'Agent Web.

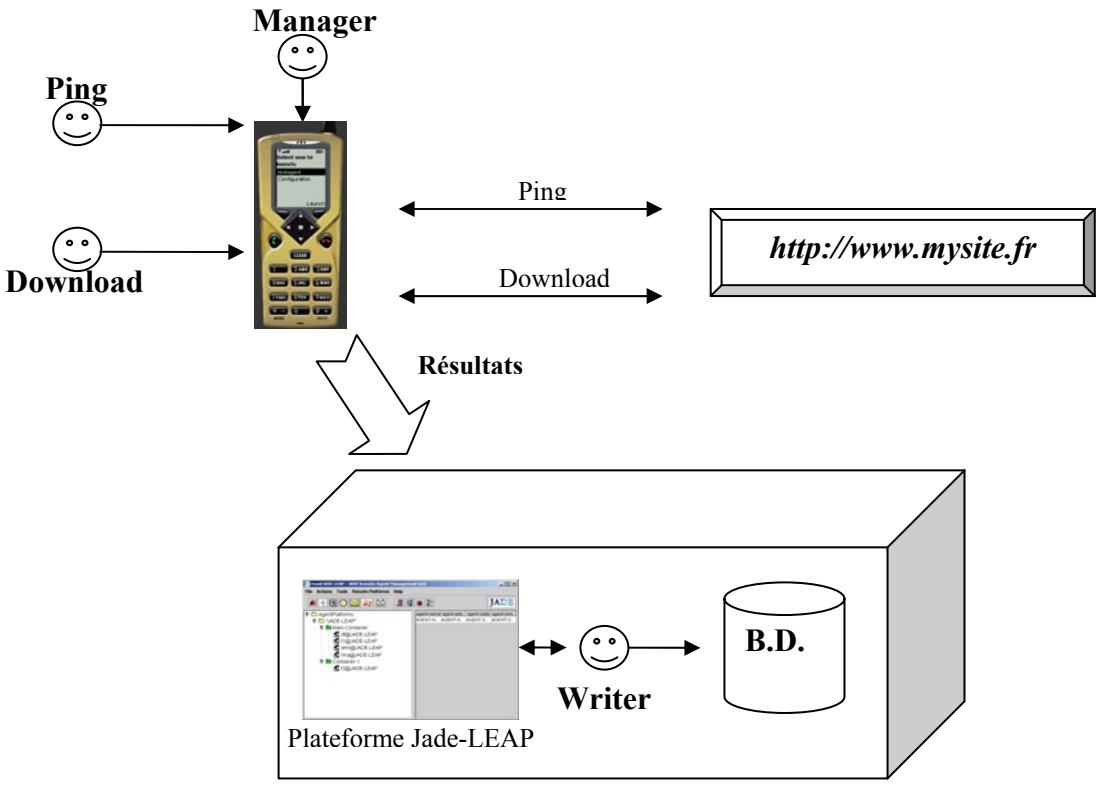


Figure 5-7 Architecture de l'**Agent Web**

Nous présentons dans la suite l'architecture et la mise en œuvre de l'**Agent Mail.** Cet agent possède une structure proche de celle de **l'Agent Web**. Pour cette raison, la présentation de la simulation de ces agents sera faite dans le même chapitre.

5.4 Architecture d'un Agent pour l'évaluation de la qualité de service pour les emails

Parmi les services multimédia qui sont proposés par les opérateurs de télécommunications nous retrouvons la réception et l'envoi des emails par un terminal mobile. En utilisant le terminal mobile, nous avons très peu d'information concernant les paramètres de qualité concernant l'email (voir la section 4.1). En s'appuyant toujours sur le principe des agents

intelligents, nous avons développé un agent qui permet l'évaluation de la qualité de service pour les emails.

Nos tests ont été réalisés sur un serveur mail de type Merak [Réf Merak] qui a été installé sur la même machine que la plate-forme Jade-LEAP (pour faciliter la tâche des agents intelligents). Le serveur Merak a été choisi pour sa facilité d'installation, de configuration et de maintenance. Celui-ci couvre tous les besoins en terme de messagerie et dispose aussi de nombreuses fonctions de sécurité. Toutes les informations concernant les utilisateurs et les mails qui ont été envoyés ou reçus se trouvent dans les fichiers **.log** du serveur. Ces fichiers ne sont consultables que par l'administrateur du réseau d'où l'intérêt de notre approche pour l'utilisateur final.

5.4.1 Agent Mail

Cet agent a été construit en s'appuyant sur notre modèle en 5 étapes de l'évaluation de la QoS et il a été implémenté sur le même principe que l'**Agent Web**.

Les principales caractéristiques que nous avons mesurées sont la disponibilité du serveur mail, ainsi que le nombre de messages envoyés et reçus par un utilisateur à une date précise. De plus nous avons la possibilité de connaître la quantité d'information qui a été transmise et envoyée dans ces mail (« le poids » en bytes des fichiers attachés). Le nombre de mails envoyés et reçus par l'utilisateur peut être ensuite comparé avec le nombre de mails envoyés et reçus par le serveur pour le même utilisateur dans la même période.

Notre **Agent Mail** est constitué de 3 agents différents :

- un agent que nous avons appelé **Mail** qui traite les fichiers **.log** du serveur mail ;
- un agent que nous avons appelé **Writer** qui a comme tâche d'écrire dans la base de données les informations qui lui ont été envoyées par **l'Agent Manager** ;
- un agent que nous avons appelé **Manager** et qui gère les 2 premiers agents.

Le principe de fonctionnement est le même que celui de l'Agent Web (voir Figure 5-9). **L'Agent Manager** lance **l'Agent Mail** qui traite les informations qui se trouvent dans les

fichiers **.log** du serveur. **L'Agent Mail** envoie les résultats à **l'Agent Manager** qui les transmet ensuite à **l'agent Writer**. Une partie des résultats (le nombre de mail envoyés, le nombre de mail reçus et le « poids » en bytes des fichiers attachés) est affichée sur le terminal mobile, tandis que les résultats complets seront mémorisés dans la base de données du serveur de l'opérateur.

Le fichier qui sera créé dans la base de données correspond à un seul utilisateur (celui qui fait le test) et les données qui se trouve dans ce fichier dépendront de la date à laquelle le test a été réalisé. Si par exemple le test concernant le mail a été fait le jour J d'un mois, les résultats retournés sont du 1er du mois jusqu'au jour J-1 du même mois. Cette option a été prise en tenant compte de ce que propose actuellement les opérateurs de télécommunications concernant la consultation du forfait pour les cartes prépayées (nous pouvons consulter le nombre de minutes qui nous restent jusqu'au jour précédent le jour de consultation).

Pour faciliter la tache des agents et pour réduire le temps de calcul, **l'Agent Mail** se trouve sur le même serveur que la plate-forme Jade-LEAP. Seul **l'Agent Manager** se trouve sur le terminal portable.

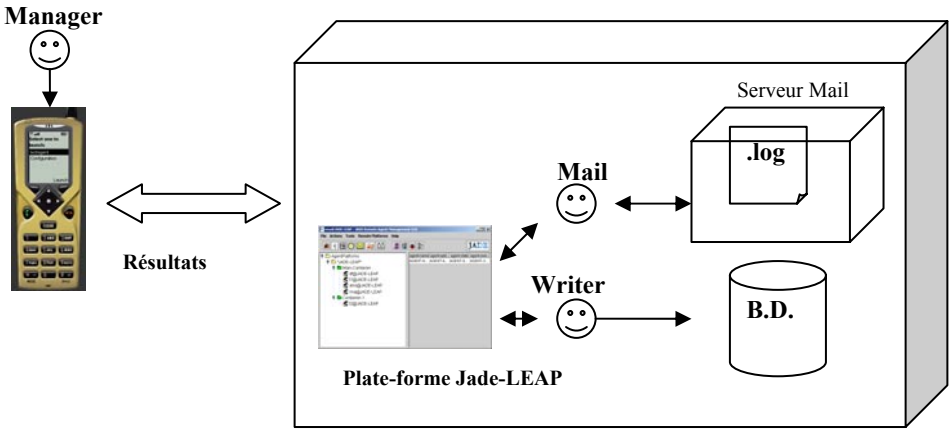


Figure 5-8 Structure de l'Agent Mail

A chaque agent correspond une ou plusieurs classes Java fonction de sa complexité. **L'Agent Manager** a les mêmes classes que **l'Agent Web**. **L'Agent Mail** a les classes suivantes : mailSwing (la classe qui gère l'interface), TestAgent (la classe qui gère la

gestion de messages), CLEANUP (la classe principale), TestGUI (la classe qui lie l'interface à l'agent) et TestCleanup (la classe qui lie la classe Cleanup à l'interface).

En tenant compte de la similarité des structures de l'Agent Web et de l'Agent Mail, nous avons choisi de fusionner ces 2 agents. Nous avons fait quelques changements au niveau de l'Agent Manager et de l'Agent Writer pour qu'ils puissent gérer les 2 agents en même temps.

Ainsi le schéma de fonctionnement de tous les agents devient :

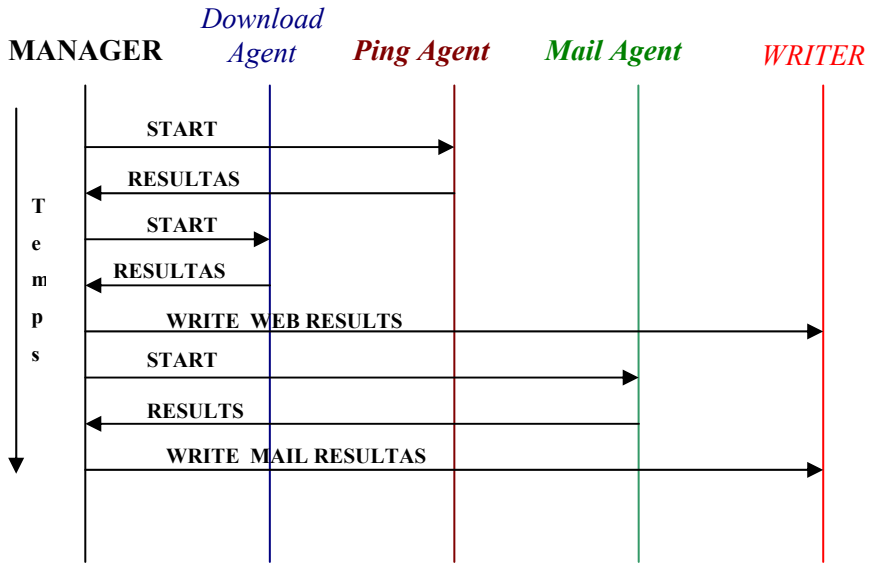


Figure 5-9 Description du fonctionnement de l'Agent Web-Mail »

Le principe de fonctionnement de l'Agent Web-Mail est identique à celui de l'Agent Web auquel nous avons ajouté l'Agent Mail. Le schéma final de l'implémentation des agents pour l'évaluation de la qualité de service pour le web et les emails sur un terminal mobile se présente de la façon suivante

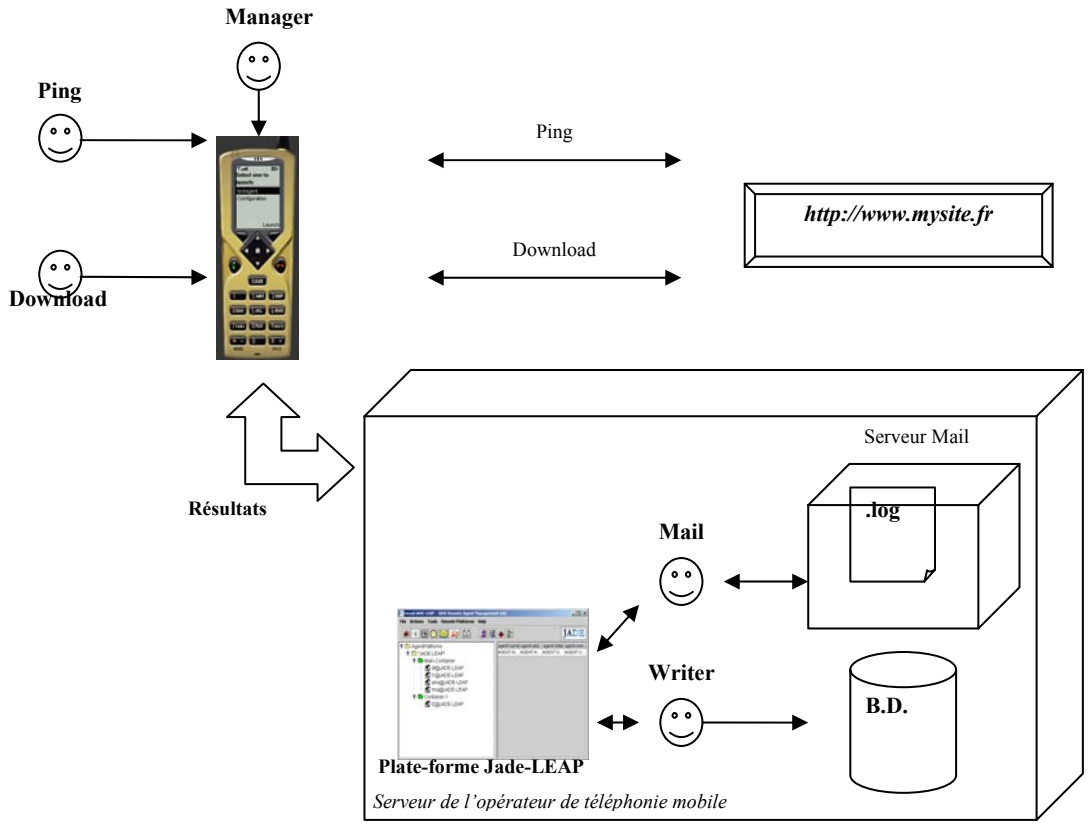


Figure 5-10 Structure de l'Agent Web et l'Agent Mail

Après avoir construit la structure des 2 agents, nous allons procéder à une simulation en utilisant le J2ME Wireless Toolkit. Le paragraphe suivant présente chaque pas de la simulation, les résultats étant commentés dans la section 5.6.

5.5 Architecture d'un Agent pour évaluation de QoS pour le streaming vidéo

5.5.1 L'Agent Vidéo

Nous avons construit l'agent « Vidéo » pour l'évaluation de la qualité de service pour le streaming vidéo. Cet agent a été construit en s'appuyant sur notre modèle en 5 étapes de l'évaluation de QoS et en tenant compte du principe de fonctionnement du streaming vidéo.

Les principaux paramètres de la qualité de service que nous avons mesurés sont présentées dans le tableau suivant :

Caractéristique vidéo	**Paramètre**	**Valeur**
Audio	*Type*	**Layer I, Layer II, Layer III**
	Nombre de trames	**X**
	Fréquence d'échantillonage (kHz)	**44.1, 48 ou 32**
	Débit binaire (kbit/sec)	**32 à 448**
Image	*Taille de la trame*	**X pixels x Y pixels< 4096x4096**
	Format géométrique	**SQUARE**
	Nombre de trames/sec	**23.976,24, 25, 29.97, 50, 59.94, 60**
	Débit binaire (kbit/sec)	**64 kbit/sec à 1,5 Mbit/sec**
	Nombre de I	**X**
	Nombre de P	**X**
	Nombre de B	**X**
	Nombre de trames	**X**
Performance réseau	*Vitesse de transfert (kbit/sec)*	**X**
	Nombre de trames transférées/sec	**X**

Tableau 5-2 Paramètres valeurs pour le service de streaming vidéo

On rappelle que les valeurs marquées par un X sont des valeurs qui dépendent de chaque séquence audio-vidéo ou dans le cas de la performance du réseau en fonction de la bande passante celle-ci.

Notre Agent Vidéo est constitué de 2 agents différents :

- un agent que nous avons appelé **Manager** qui gère l'interface, la connexion au serveur et réalise l'évaluation de la qualité de service pour différents fichiers MPEG-1 spécifiés ;

- un agent que nous avons appelé **Writer** qui a comme tâche d'écrire dans la base de données les informations qui lui ont été envoyées par l'**Agent Manager**.

Le principe de fonctionnement de l'Agent Vidéo ressemble à celui de l'Agent Web sur la partie interface et accès au site Internet demandé par l'utilisateur ainsi que sur la partie de l'**Agent Writer**. Nous avons décomposé le fonctionnement de l'**Agent Vidéo** en plusieurs pas d'exécution comme suit :

- l'**Agent Manager** récupère l'adresse du serveur web introduite par l'utilisateur final sur son mobile ;

- le même agent lance un nouveau processus («thread» en anglais) où sera réalisé l'ensemble des opérations permettant de garder l'interface active ;

- l'**Agent Manager** effectue la connexion au serveur web pour retrouver le fichier MPEG à télécharger ;

- la mémoire tampon (buffer) du téléphone mobile (dans notre cas le simulateur J2ME) sera remplie ;

- l'évaluation de la qualité de service sur le fichier MPEG est réalisée par **l'Agent Manager**[29] ;

[29] La présentation du mode d'évaluation de la qualité de service pour des fichiers MPEG sera faite dans un paragraphe à part.

- le même Agent récupère les résultats et les envoie à l'**Agent Writer** qui a comme tâche de les afficher sur le mobile et dans la base de données de l'opérateur (l'**Agent Writer** fonctionne sur le même principe que l'Agent Writer de l'**Agent Web**).

La partie image et la partie audio des fichiers multimédia ont été traitées de manière différente. Tout d'abord nous avons remarqué que sur un fichier multimédia, la partie audio ne démarre pas forcément en même temps que la partie image. Une analyse audio et image de chaque séquence audio-vidéo n'est pas envisageable à cause du temps de calcul trop élevé.

Afin de pouvoir disposer des mesures quantitatives sur les fichiers multimédia diffusés nous avons implémenté un analyseur syntaxique compatible avec la norme MPEG-1. Le but de ce logiciel est de fournir, pendant la diffusion des données audio-vidéo, des paramètres spécifiques à chacun, tel que la dimension de l'image, la fréquence d'échantillonnage audio, ou bien des paramètres spécifique a la séquence diffuse tel que le nombre des trames vidéo. La norme MPEG-1 prévoit des chaînes binaires spécifiques, appelées *start code* ou *end code* afin de pouvoir naviguer facilement dans le flot multimédia. Ainsi, pour extraire les informations permettant la quantification de la qualité il n'est pas nécessaire d'effectuer le décodage complet du flux, mais une analyse syntaxique qui exploite les *start* et *end code*.

La procédure de l'analyseur syntaxique est la suivante :

1. identification de la variable *pack_start_code* (**=000001BA**) qui détermine le début d'une couche pack ;

2. remplissage de buffer avec le contenu d'un pack entier (délimité entre deux *pack_start_code*) ; les pas suivants seront réalisés dans la structure buffer ;

3. identification de la variable system_header_start_code (**=000001BB**) qui détermine le début d'un header système ;

4. avancement de 64 bits et lecture des prochaines 8 bits qui représente le *stream_id*.

5. identification de la variable *packet_start_code_prefix* (=**000001**) qui détermine le début d'un paquet et lecture de la variable *stream_id*. Par composition entre *packet_start_code_prefix* et *stream_id* il est obtenu le *packet_start_code*. Cette variable permet de connaître la nature de flux élémentaire présent dans le paquet courant : audio, vidéo ou privé. S'il s'agit de l'information vidéo la procédure continue avec le pas 6. S'il s'agit de l'information audio la procédure passe au pas 13. Si non elle retourne au pas 3. Avancement jusque au début de la section *packet_data_byte*.
6. identification de la variable *sequence_header_code* (=**000001B3**) ;
7. lecture des paramètres : (*horisontal_size* 12 bits, vertical_size 12 bits, pel_aspect_ratio 4 bits, picture_rate 4 bits et bit_rate 18 bits) ;
8. identification de la variable *group_start_code* (=**000001B8**) ;
9. identification de la variable *picture_start_code* (=**00000100**) ;
10. avancements de 10 bits ;
11. lecture de la variable picture_coding_type. Si cette variable est égal à 1, le paramètre « Intra » est incrémenté, si elle est 2 le paramètre « Prédictif » est incrémenté et si elle est égal a 2 le paramètre « Bidirectionnel » est incrémenté ;
12. vérification de l'une des trois variables: *group_start_code*, *picture_start_code* ou *sequence_end_code*. Si la première est retrouvée la procédure reprend le pas 9, si c'est la deuxième, le pas 10. Si la troisième variable est retrouvée une vérification supplémentaire sur les variables *iso_11172_end_code* et *pack_start_code* est effectuée. Dans le premier cas la procédure s'arrête, dans le deuxième elle retourne au pas 3 ;
13. vérification de la chaîne binaire '**1111 1111 1111**' permettent la synchronisation ;
14. vérification de l'idéntificateur *ID* (1bit) ;
15. vérification de l'idéntificateur *Laye*r. Si égal 1, le paramètre « couche » est 3, si égal 2, le paramètre « couche » est 2 et si égal 3 le paramètre « couche » est 1 ;

16. avancement d'un bit ;

17. lecture de la variable *bit_rate_index* (4 bits) permettant d'identifier la variable «débit binaire»;

18. lecture de la variable *sampling_frequency* (2 bits) permettant d'identifier la variable «fréquence d'échantillonnage» ;

19. vérification de l'une des deux variables : *syncword* ou *sequence_end_code*. Si la première est retrouvée la procédure reprend le pas 14. Si la deuxième variable est retrouvée une vérification supplémentaire sur les variables *iso_11172_end_code* et *pack_start_code* est effectuée. Dans le premier cas la procédure s'arrête, dans le deuxième elle retourne au pas 3.

A la fin de cette procédure nous retrouvons les paramètres de la qualité de service de streaming vidéo présentés dans le Tableau 5-2.

5.6 Simulation des Agent Web, Mail et Vidéo sur la plateforme JADE-LEAP à l'aide du J2ME Wireless Toolkit

Chaque pas de la simulation est présenté dans la Figure 5-11

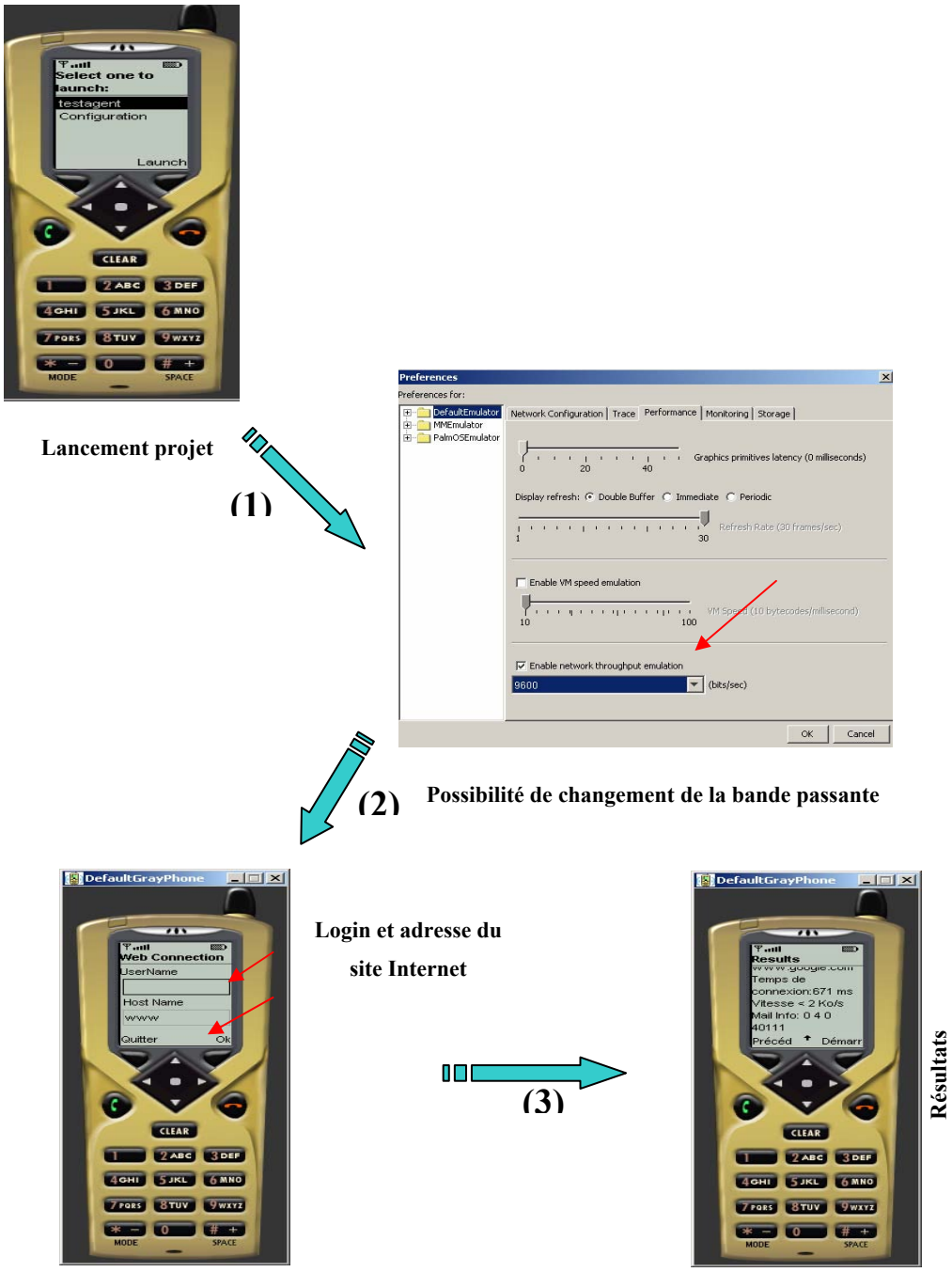


Figure 5-11 Simulation des Agent Web, Mail et Vidéo sur la plateforme JADE-LEAP à l'aide du J2ME Wireless Toolkit

Tout d'abord il faut lancez le Toolkit, particulièrement le programme appelé *KToolbar*.

Ensuite il faut créer un projet à l'aide du bouton Create Project (dans notre cas « testagent »), ainsi que le nom de la classe *MIDlet* principale qui doit être exécutée pour démarrer l'application. Le Toolkit va créer alors un répertoire de projet portant le nom que nous avons indiqué.

(1) Avant d'appuyer sur « Launch » pour lancer la plateforme Jade-LEAP, nous avons la possibilité de simuler différents types de réseaux (il s'agit d'une bande passante entre 12 et 112 kbit/s) (voir Figure 5-11). La simulation telle qu'elle a été implémentée sur notre système s'exécute beaucoup plus rapidement que sur un système réel. [Réf Wireless Toolkit]. En fait les temps de connexion et les vitesses de transfert correspondent à la puissance de calcul de la machine sur laquelle la plateforme a été installé et de la connexion réseau de la machine vers le réseau Internet.

(2) Une fois la plateforme lancée, l'utilisateur doit introduire son login et l'adresse du serveur sur lequel il souhaite faire le test (l'adresse doit être exacte).

(3) Après l'exécution du test, l'adresse du dernier serveur est affichée sur le mobile ainsi que la le temps, la vitesse de connexion, le nombre de mails envoyés, reçus ainsi que la taille des fichiers attachés.

Pour avoir plus de détails concernant toutes les opérations sur des différentes serveurs, l'utilisateur a la possibilité de consulter le fichier log de la base de données du l'opérateur de téléphonie mobile. Ce fichier peut être consulté par l'intermédiaire du site Internet de l'opérateur.

5.7 Forme de résultats obtenus

Nous avons fait différentes simulations sur l'accès web, le mail et le streaming vidéo pour 3 types de réseau (9.6 kbit/sec, 28.8 kbit/sec, 112 kbit/sec). En fait les résultats obtenus montrent bien que le choix du réseau reste fictif et dépend de la connexion Internet de la machine sur laquelle la plateforme a été installée.

Nos résultats (voir Annexes) montrent :

- la capacité de notre système multi-agents d'évaluer la QoS pour un mode de classique ou par paquet ;
- l'évaluation passe par des agents de utilisateur directement au service à l'essai ;
- un utilisateur est complètement indépendant du processus d'évaluation ;
- la stabilité et l'exécution de ce système multi-agents ;
- les utilisateurs reçoivent des résultats complètes ;
- l'intérêt d'évaluer la QoS par l'utilisateur par l'intermédiaire d'une interface facile à utiliser.

Cependant, l'utilisateur n'acceptera pas un ensemble de données techniques telles que celles que nous avons obtenues. Il a besoin seulement d'un nombre comme le MOS[30] utilisé en téléphonie IP. Malheureusement l'absence de contrats de service et de vraies plates-formes qui proposent des services multimédia nous metteent dans l'incapacité de réaliser entièrement la dernière étape dans notre processus d'évaluation de la qualité de service.

5.8 Résumé du chapitre

Au cours de ce chapitre nous avons décrit notre implémentation de l'évaluation de la qualité de service pour l'accès web, le courrier électronique et le streaming vidéo. Cette implémentation a été réalisée sur la plateforme Jade-LEAP. L'architecture de la mise en œuvre des Agents a été présentée en détails. L'Agent web et l'Agent Mail ont été crées sur la même structure en tenant compte des services mobiles actuels (un utilisateur qui possède un abonnement GPRS peut avoir accès au web et à ses mails à condition d'avoir un adresse électronique donnée par son opérateur mobile). L'architecture de L'Agent Vidéo ainsi que

[30] Voir section 3.2.5

les étapes de la simulation ont été traitées dans des chapitres à part. A travers les résultats obtenus nous avons prouvé que les agents peuvent servir à une évaluation automatisée de la qualité de service pour différentes applications et services avec une interface proche du téléphone mobile actuel.

6 Conclusion et travail futur

Ce dernier chapitre résume le travail présenté dans les chapitres précédents. Il examine l'approche théorique que nous avons proposée en ce qui concerne l'évaluation de la qualité de service ainsi que son 'implémentation à l'aide des Agents Intelligents. Le travail futur et les implications potentielles ainsi que les problèmes rencontrés et les problèmes non résolus sont présentés.

6.1 Résumé de la thèse

On assiste actuellement au développement de l'offre des services de communications fixes ou mobiles. Les opérateurs et les fournisseurs de service proposent non seulement des services liés à la voix, mais également des services multimédia qui exploitent les ressources croissantes des réseaux. Les futures normes qui ont été annoncées (UMTS) permettent aux utilisateurs l'accès à une gamme variée de services passant par un niveau de qualité de service contractuel. Introduire l'évaluation de la QoS par l'utilisateur final dans un système implique de :

- construire un cadre généralisé de la qualité de service ;
- donner des spécifications de la qualité de service par rapport aux différentes applications ;
- tracer la qualité de service par rapport aux ressources techniques ;
- faire une évaluation de la qualité de service entre des données réelles (terrain) et les données contractuelles.

Pour le premier point, aucun opérateur mobile ne propose de cadre général de la qualité de service. Les normes internationales ne sont pas encore suffisantes.

Les spécifications QoS sont des conditions données en termes de paramètres qui indiquent ce dont l'utilisateur a besoin. Les spécifications QoS sont différentes pour chaque couche du système (voir Figure 1-1) et elles seront utilisées pour la configuration des mécanismes de qualité de chacune de couches.

Ces spécifications sont suivies à l'aide de paramètres quantitatifs de QoS concernant les diverses couches du système qui peuvent être surveillés et commandés. Chaque paramètre

QoS peut être vu comme une variable avec une valeur de seuil qui doit se trouver dans les contrats de service et qui est évalué avec les données réelles du système.

Les conditions de la qualité peuvent et doivent évoluer dans le temps. Ceci implique une évolution des paramètres QoS. Si, par exemple, le niveau du service demandé ne peut pas ou ne peut plus être fourni, l'opérateur doit demander à l'utilisateur si un certain niveau de dégradation peut être acceptable. La présence d'un contrat de service ouvert et extensible peut faciliter cette approche, mais la qualité de service pour un opérateur ou un fournisseur de service ne doit pas rester une problématique d'ordre technique. En général, le client ne connaît pas et ne s'intéresse pas à la façon dont le service lui est accessible. Tout ce qui l'intéresse est son bon fonctionnement et la satisfaction du contrat de qualité.

En partant du principe qu'il est possible de trouver un procédé d'évaluation de la qualité de service simple et commun pour les utilisateurs, que le processus d'évaluation peut être automatisé, cette thèse propose un cadre d'évaluation de QoS adapté aux intérêts et aux besoins des utilisateurs.

Ce cadre complète les cadres existants en termes de management et évaluation QoS. Par rapport au Tableau 2-11 nous avons rajouté une colonne qui représente notre apport aux cadres actuels.

	ISO 9000	**ISO 9126**	**I.350**	**X.641/ X.642**	**X.900**	**EG 202 009**	**Cadre proposé dans la thèse**
Orientation du cadre							
Focalisation sur l'utilisateur final	++	+	++	-	++	++	++
Evaluation versus gestion	++	+	++	+	-	+	++
Orientation vers l'application	+	+	--	+	++	++	++
Méthode d'évaluation	0	0	-	0	0	+	++
Les caractéristiques de la Qualité							
La perception de l'utilisateur final	-	+	++	--	++	++	++
Focalisation sur quelques caractéristiques	--	--	-	--	-	++	++
Les propriétés de la méthode d'évaluation							
L'appréhension	0	0	0	0	0	+	--
L'autonomie	0	0	0	0	0	++	+
L'automatisation	0	0	0	0	0	0	++
La facilité d'utilisation	0	0	--	0	0	++	+
L'évaluation des résultats							
La comparabilité	+	0	+	0	0	+	++
La reproductibilité	+	0	+	0	0	+	++

Tableau 6-1 Les normes internationales examinées et complétées concernant QoS

++: très bon, +: bon, 0: non pertinent ; -: pauvre ; --: très pauvre

Le cadre que nous avons développé dans un environnement distribué couvre les intérêts des utilisateurs pour la QoS. Il adresse les paramètres de la qualité d'un service qui peuvent être directement observés et mesurés au point où l'utilisateur a la possibilité de consulter ce service. Les paramètres sont choisis par l'utilisateur final et le processus a été automatisé le plus possible. L'utilisation des protocoles largement répandus comme le SMTP pour le mail et le http pour l'accès web et le streaming vidéo nous a facilité la tâche concernant la communication entre l'utilisateur et les autres systèmes impliqués dans le processus d'évaluation. Les résultats obtenus sont accessibles à n'importe quel moment sur le serveur de l'opérateur ou du fournisseur de service et peuvent être comparés avec les données contractuelles. Aujourd'hui, malheureusement ces données n'existent pas, ce qui nous mene aux problèmes rencontrés et les points non résolus de la thèse.

6.2 Problèmes rencontrés et points non résolus de la thèse

Au commencement de cette thèse (fin 1999), tout le monde annonçait l'arrivée d'un réseau UMTS pour l'année 2002. Le retard pris dans le développement des technologies, les coûts énormes de la mise en place d'un tel réseau ont empêché le déploiement de l'UMTS dans les délais initialement prévus. Ceci m'a beaucoup pénalisé puisque d'une part je n'ai pas pu réaliser de tests sur des services existants sur le réseau[31] et d'autre part les utilisateurs (ma cible) n'existent pas encore.

Aujourd'hui avec l'arrivé du GPRS quelques services comme l'accès web et le mail se mettent en place sur les mobiles. Le nombre de clients par contre reste assez limité. Une des explications est l'échec du WAP qui a été annoncé comme le nouveau « Internet Mobile ». Le premier facteur de l'échec est la pauvreté des applications développées par les opérateurs. Les services le plus souvent proposés sur les portails WAP (météo, bourse, programmes de cinéma, etc.) existent déjà sous forme de serveurs vocaux qui apparaissent plus adaptés à une consultation via un téléphone mobile. Les envois de messages d'alertes pour des applications en finance, en santé ou pour fixer des réunions de travail peuvent se faire sous forme de SMS[32] et ne nécessitent pas un mobile WAP. Le modèle choisi par les opérateurs pour le lancement du WAP en Europe, en porte-à-faux avec les principes de l'Internet, n'a pas

[31] Au laboratoire CRITIC nous avons travaillé sur un projet commun avec Bouygues Télécom (un des opérateurs mobile français) qui nous a promis une plateforme de test pour des services multimédia. Malheureusement cette plateforme n'est toujours pas disponible.
[32] SMS : Short Message System

favorisé la créativité, en conférant aux seuls portails WAP des opérateurs la responsabilité d'inventer les applicatifs qui auraient pu assurer le succès de ce standard. Les applications les plus porteuses sont autour de la consultation et de l'envoi de mails, mais il reste quand même à résoudre les problèmes liés au nombre et à la taille des mails.

Avec le GPRS la consultation du web sera plus facile, mais les capacités de l'écran du téléphone mobile restent encore limitées. Les constructeurs de mobiles font de plus en plus d'efforts pour l'ergonomie et la taille des mobiles, mais le prix reste assez élevé (un portable avec un grand écran en couleur coûte autour de 600-700 euros).

Les applications vidéo (vidéoconférence, streaming vidéo) demandent des bandes passantes assez importantes et avec les réseaux existants en Europe leur développement est quasiment impossible. Dans les pays asiatiques les opérateurs (NTT DoCoMo) proposent des services multimédias à travers les portables.

Cette situation nous a amené à utiliser des simulations des applications mobiles. L'utilisation de la plateforme jade LEAP et du J2ME Wireless Toolkit nous a permis de faire ce type de simulations. Malgré ces avantages la plateforme Jade LEAP possède aussi quelques inconvénients. Premièrement aucune interface de développement n'est disponible pour cette plateforme. A cause de cette lacune, l'implémentation demande beaucoup d'efforts. Elle nécessite une bonne connaissance des classes et différents services offerts. Cette lacune a été comblée par le Java Wireless Toolkit qui offre des possibilités assez proches des mobiles actuels. Même si cette application nous offre la possibilité de choisir parmi plusieurs types de bande passante (voir section 5.6) celle-ci reste fictive puisque nous ne changerons pas de type de réseaux. Les résultats que nous avons obtenus ne correspondent pas à des réseaux avec différentes bandes passantes, mais à notre propre type de réseau et surtout à notre type de connexion Internet.

Un autre inconvénient de la plateforme : une faute d'orthographe (nous rappelons qu'à travers l'écran du portable nous allons introduire manuellement le login et l'adresse URL du site sur lequel nous allons faire les tests) peut provoquer le blocage de la plateforme. Dans un cas réel cela peut arriver couramment.

Les tests ont été réalisés avec un seul utilisateur, ce qui ne sera pas le cas sur un réseau d'accès mobile. Cette capacité n'a pas encore été explorée et il se peut qu'il soit impossible de développer plusieurs accès à la fois sur cette plateforme.

Pour le streaming vidéo, le code nécessite des améliorations pour gagner en temps d'exécution. L'évolution de la norme MPEG 1 vers le MPEG 4 et MPEG 7 obligera à faire des changements au niveau du code. La norme MPEG 21 peut aussi apporter une solution envisageable à notre problématique. Notre but n'était pas de réaliser des programmes très optimisés, mais d'avoir des applications pouvant montrer la faisabilité technique de l'évaluation de la qualité de service.

Un autre point non résolu de cette thèse est le manque d'études de marché auprès des utilisateurs. A l'heure actuelle le nombre d'utilisateurs des services que nous proposons pour le test est très faible en Europe (1,2 % de la totalité des utilisateurs de mobiles). De plus comme nous l'avons déjà précisé les services de streaming vidéo n'existent pas.

Il est important de préciser aussi que dans la démarche de l'évaluation de la qualité de service et notamment pour établir la procédure en 5 étapes nous avons souvent pris la place d'un utilisateur. En réalité il faut donner la possibilité aux utilisateurs finaux de choisir eux-mêmes les services et donc les paramètres correspondants qu'ils souhaitent tester. Ceci implique des développements beaucoup plus complexes, mais tant que les services n'existent pas il est difficile de juger.

6.3 Travail futur

Dans un premier temps, des tests avec plusieurs utilisateurs sur la plate-forme seront faits. Au cas où cette solution serait inexploitable il faudra changer la plateforme. Une amélioration du code de programmation peut être faite également. Le code reste ouvert et extensible. Une migration vers d'autres services (video-on-demand et ftp) sera possible. Elle implique la présence sur le marché des services et des contrats de services.

La solution avec des Agents Intelligents peut être modifiée. La programmation de « l'intelligence » de ces agents reste une question relative. L'utilisateur n'est pas impliqué dans cette démarche, son rôle est juste de réaliser les tests. Avec le modèle actuel nous avons imposé un certain nombre de paramètres qui peuvent être changés et améliorés à la demande de l'utilisateur.

Dans les futures normes internationales les paramètres de la qualité de service seront assez précis et faciliterons la tâche dans le choix du développement de la procédure d'évaluation.

Une étude de marché sera faite auprès des utilisateurs de mobiles actuels pour donner éventuellement de nouvelles directions à cette procédure. Pour que cette étude de marché soit réalisable il faut avoir accès aux bases de données clients des opérateurs.

Une analyse financière peut être envisagée pour connaître les coûts du développement d'un outil pour le client d'un côté et les opérateurs et les fournisseurs de services de l'autre.

6.4 Implications potentielles

Cette thèse apporte une nouvelle orientation dans le domaine de la qualité de service. Tout en restant à une extrémité de chaîne de la qualité de service de bout en bout, l'impact de l'évaluation de QoS est assez important.

Le management de la qualité de service s'adresse plus aux opérateurs, mais à partir des résultats fournis par les utilisateurs il peut :

- contrôler la qualité de service ;
- employer ou renégocier les accords de service ;
- réduire au minimum ses coûts opérationnels et d'entretien ;
- donner des priorités et programmer les activités de maintenance selon le niveau de l'impact sur le service ;
- contrôler les services indépendamment des fournisseurs et de technologies.

En 1997, les clés de la différenciation entre les opérateurs étaient la qualité de l'appel, les tarifs, la couverture, le service au client dans l'usage de son mobile et les services à valeur ajoutée. Les tendances du marché actuel et surtout pour les futurs réseaux tournent autour de la fiabilité, l'adaptabilité et l'intelligence du système, la localisation liée au service bidirectionnel (client-réseau, réseau-client), l'identification sécurisée et surtout la qualité de service de bout en bout. Tout ceci se sent dans le comportement du consommateur, le téléphone mobile est présent partout. Après l'avoir accepté, l'utilisateur final formule de nouveaux besoins. Selon Daniel Salingre dans [Réf HOURS] du CS Télécom [Réf CS Télécom]« les constructeur et les opérateurs suivent le mouvement, ils innovent très peu. Par contre s'ils voient qu'un produit marche, ils s'y mettent ».

Aujourd'hui les opérateurs sont plus intéressés par la préservation de leurs équipements par la protection et l'augmentation de leur clientèle aux dépends d'autres services plus où moins avantageux. L'apparition de contrats entre les clients et les opérateurs ou les fournisseurs de service peut changer radicalement cette situation. Notre procédé d'évaluation peut apporter une réponse aux besoins des clients utilisateurs.

BIBLIOGRAPHIE

[768] J. Postel, User Datagram Protocol, 28 August 1980

[959] J. Postel and J. K. Reynoids, File Transfer Protocol, Network Working Group, October 1985

[1521] N. Borenstein and N. Freed. MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing thé Format of Internet Message Bodies. September 1993

[1889] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. January 1996

[2039] C. Kalbfleisch. Applicability of Standards Track MIBs to Management of Worid Wide Web Servers. November 1996

[2326] Network Working Group : H. Schulzrinne (Columbia U) ; A. Rao (Netscape); R. Lanphier (RealNetworks) April 1998

[2594] H.Hazewinkel, C.Kalbfleisch and J.Schoenwalder, Definitions of Managed objects for www services, May 1999

[AFNOR] Association Française de Normalisation http://www.afnor.fr/portail.asp

[AgentBuider] AgentBuilder http://www.agentbuilder.com

[AgentTool] AgentTool http://en.afit.af.mil/ai/agentool.htm

[Akamai] http://www.akamai.com

[Alper97] Alper Caglayan et Colin Harrisson - Les Agents - Editions InterEditions 1997

[Apache] http://httpd.apache.org/

[ART 2000] http://www.art-telecom.fr/ : Enquête 2000 sur la qualité de service des réseaux de téléphone mobile en France

[ASG] http://asg.web.cmu.edu/cyrus/email/standards.html

[Blair] G.Blair , Open Distributed Processing and Multimedia.Addison-Wesley, ISBN 0-201-17794-3, 1997

[BOGEN] Manfred BOGEN , PhD Desertation " A Framework for Quality of Service Evaluation in Distributed Environnements, 2001"

[Bonnet] « Les Agents intelligents sur l'Internet », Ecole Polytechniques de l'Université de Nantes

[Bradner] S.Bradner « The Internet Standard Process – Revision 3, Octobre 1996 »

[Brett] R.Brett, S.Bradner and G.Parson Collaboration entre IETF et ITU-T, Octobre 1998

[CAG97] Alper Caglayan et Colin Harrisson – « Les Agents » - Editions InterEditions 1997

[CIS] http://www.cis.ohio-state.edu/~jain/refs/ipqs_ref.htm#books

[Columbia] http://ctr.columbia.edu/~sbl/notes.html

[CS Télécom] http://www.icetact.tcd.ie/icetact/cstelecom/pres_cs.html

[David97] J.F. David « Les agents intelligents : une question de recherche », revue SIM (éditions ESKA), volume 2, 1997

[Delb] Bruno Delb : J2ME Applications Java pour les terminaux mobiles,Editions Eyrolles 2003 , chapitre 1 page 11

[DotNet] http://www.dotnetguru.org/articles/J2MEvsSDE/J2MEvsSDE.htm

[E.800] ITU-T Recommandations. Terms and Definitions Related to the Quality of Service and Network Performance including Dependability. August 1994

[EG 202 009] ETSI 202 009 User group ; Quality of Service Telecom, December 2001

[Engel] Fred Engel. Thé Rôle of Service Level Agreements in thé Internet Service Provider Industry. In: International Journal of Network Management 9, 299-301 (1999), John Wiley Sons, Ltd., 1999

[ETSI] European Telecommunications Standards Institute web page http://www.etsi.org

[FIPA] http://www.fipa.org

[FIPA23] http://www.fipa.org/specs/fipa00023/

[FOUQ] B. FOUQUET « Gestion de la qualité de service, Edition Eyrolles, 2000, ISBN 2212092407 »

[GARN] Garneau&Sylvain Delisle « Programmation orientée-agent : évaluation comparative d'outils et environnements » JFIADSMA 2002 28-30 octobre, Lille, France

[GODLE/LAGRA] Lagrange, Xavier / Godlewski, Philippe / Réseaux GSM-DCS , 4e éd, Paris : Hermès, 1999, N° ISBN 2-7462-0028-7

[Gouv] http://www.telecom.gouv.fr/telecom/car_umts.htm

[Grass] http://www.grasshopper.de/

[H.NWA] http://Agents.umbc.edu/introduction/hn-dn-ker99.htm

[HARDY] William C. Hardy « Measurement and Evaluation of Telecommunications Quality of Service, QSDG Magazine, 2001 ISBN 0471499579 page 3 »

[Hervé] Hervé Bénoit La télévision numérique MPEG1, MPEG2 système européen DVB, Edition Dunod, ISBN 2 10 003866 4 page 47

[HOURS] Manuelle HOURSIN, Gonul BASODA, Caroline VEDOVATO : QOSMIC Comment gérer la qualité de service page 11

[I.350] Les Aspects Généraux Relatifs à Qualité de Service et les Performances des Réseaux Numériques, Mars1993

[INFRES] http://www.infres.enst.fr/~beker/QoS2.htm

[INRIA] http://www.inria.fr/rapportsactivite/RA2000/armor/armor_Toc.html

[IS0 8402] ISO. Quality management and quality assurance; Vocabulary. 1991

[ISO 9000] ISO. Quality management and quality assurance standards. Parts 1-4. 1994

[ISO 9000 :2000] ISO. Quality management and quality assurance standard. 2000

[IS0 9001] ISO. Quality Systems - Model for quality assurance in design/development, production, installation and servicing. 1994

[IS0 9002] ISO. Quality Systems - Model for quality assurance in production, installation and servicing. 1994

[IS0 9003] ISO. Quality Systems - Model for quality assurance in final inspection and test.1994

[IS0 9004] ISO. Quality management and quality system elements - Guidelines. Parts 1-4.1994

[IS0 9126] ISO/IEC. Information technology - Software product Evaluation - Quality characteristics and guidelines for their use. 1991

[ISO 8402] ISO. Quality management and quality assurance; Vocabulary. 1991

[ISO/IEC 11172-1] Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s -- Part 1: Systems

[ISO/IEC 11172-2] Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s -- Part 2 :Video

[ISO/IEC 11172-3] Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s -- Part 3 :Audio

[ISO/IEC 9126] ISO/IEC. Information technology - Software product evaluation - Quality characteristics and guidelines for their use.1991

[ISO/IEC 9126-1] Génie du logiciel -- Qualité des produits – Partie 1: Modèle de qualité

[ISO/IEC 9126-2] Génie du logiciel -- Qualité des produits -- Partie 2: Mesures externes

[ISO/IEC 9126-3] Génie du logiciel -- Qualité des produits -- Partie 3: Mesures internes

[ISO/IEC 9126-4] Génie du logiciel -- Qualité des produits -- Partie 4: Quality in use metrics

[ITU-T00] The International Communication Union, page web http://itu.int/ITU-T

[J2ME] http://www.sun.com

[J2MEtoolkit] http://java.sun.com/products/j2mewtoolkit/

[JADE] http://sharon.cselt.it/projects/jade

[JPEG] ISO/IEC 10918

[LEAP] http://leap.crm-paris.com/

[Leydekkers96] P. Leydekkers and Valérie Gay. ODP View on Quality of Service for Open Distributed Multimédia Environments. In: Quality of Service - Description Modelling and Management. Jan de Meer, Andréas Vogel (Eds.), Proceedings ofthe 4. International IFIP Workshop on Quality of Service IWQoS96, Paris, March 6-8, 1996, ISBN 3-88457-282-2, pg. 31-43, 1996

[MadKit] Madkit http://www.madkit.org

[MASI94] L. Besse, L. Dairaine, L. Fedaoui, W. Tawbi, and K. Thai. Towards an Architecture for Distributed Multimédia Application Support. In: Proceedings of the International Conférence on Multimédia Computing and Systems, Boston, May 1994

[Merak] http://www.icewarp.com

[Midp4Palm] http://java.sun.com/products/midp4palm/index.html

[MPEG-1] ISO/IEC. Coding of moving pictures and associated audio for digital storage media at up to 1,5 Mbit/s. International Standard 11172, Parts 1-5, 1993-1995

[MPEG-2] ISO/IEC. Generic coding of moving pictures and associated audio information. International Standard 13818, Parts 1-6, Part 9, 1994

[MPEG-4] ISO/IEC. Coding ofaudio-visual objects. International Standards 14496, Parts 1-6, 1999

[MPEG 21] http://mpeg.telecomitalialab.com/standards/mpeg-21/mpeg-21.htm

[NAHRS] Klara NAHRSTEDT, Nalini Venkatasubramanian and Klara Nahrstedt, An Integrated Metric for Video QoS, Proceedings of ACM Multimedia '97

[Nahrstedt95] Klara Nahrstedt. An architecture for end-to-end quality of service provision and its expérimental validation. Ph.D. Dissertation. Computer and Information Sciences, University of Pennsylvania, USA, 1995

[OMG-AWG] Object Management Group – Agent Working Group http://www.objs.com/isig/Agents.html

[Radu, Salgues] Augustin Radu, Bruno Salgues « Rapport sur la technique UMTS et les services », 2001

[Real Media] http://www.realnetworks.com

[Shenker] S.Shenker « Network Element Service Specification template, septembre 1997 »

[SHOHAM] Shoham Y, Agent- Oriented Programming, Artificial Intelligene, Vol. 60, No 1, p. 51-92, Mars 1993

[TCP] 1180TCP/IP Tutorial

[TILAB] http://www.telecomitalialab.com/

[Windows] Media http://www.microsoft.com

[Wireless] Toolkit Wireless Toolkit Version 1.0.4 Java 2 Platform, Micro Edition User's Guide page 38

[X.400/84] ITU-T Recommendations X.400-X.430. Data communication networks: message handling Systems. Recommendations X.400-X.430, Malaga-Torremolinos, October 1984

[X.400/88] ITU-T Recommendations X.400-X.430. Data communication networks: message handling Systems. Melbourne, November 1988

[X.641] Technologie De l'Information - Qualité de service : La structure

[X.642] Technologie de l'information- Qualité de service- Le guide des méthodes et des mécanismes

[X.901] ITU-T Recommendation. Information Technology - Open distributed processing -Référence Model: Overview. OMG, 1995-98.

[X.902] ITU-T Recommendation. Information Technology - Open distributed processing -Référence Model: Foundations. OMG, 1995-98.

[X.903] ITU-T Recommendation. Information Technology - Open distributed processing - Référence Model: Architecture. OMG, 1995-98.

[X.904] ITU-T Recommendation. Information Technology - Open distributed processing - Référence Model: Architectural semantics, OMG, 1995-98.

[XRM94] A.A. Lazar, S. Bhonsie, and K.S. Lim. A Binding Architecture for Multimédia Networks. In: Proceedings ofthe COST237 Conférence on Multimédia Transport and Teleservices, Vienna, Austria, 1994

[ZEUS] http://www.labs.bt.com/projects/agents/zeus/

[VirtualDub] http://www.virtualdub.org

[W3] http://www.w3.org

Annexes

Annexe 1 Résultats de l'évaluation du mail

Date	Heure	Adresse mail utilisateur	Nombre de mails reçus	Nombre de mails envoyés	Taille (nombre d'octets) des fichiers attachés reçus	Taille (nombre d'octets) des fichiers attachés envoyés
30-oct-03	02:09:45	augustin@testevry.int-evry.fr	3	3	1279045	1014725
30-oct-03	02:13:14	augustin@testevry.int-evry.fr	4	3	1279045	1014725
30-oct-03	02:26:38	augustin@testevry.int-evry.fr	4	4	1279045	1014725
06-nov-03	11:40:39	augustin@testevry.int-evry.fr	4	0	40111	0
03-déc-03	14:53:49	augustin@testevry.int-evry.fr	0	0	0	0
03-déc-03	14:56:24	augustin@testevry.int-evry.fr	0	1	0	0
03-déc-03	14:56:24	augustin@testevry.int-evry.fr	1	1	1200	0

Annexe 2 Résultats de l'évaluation de l'accès web

L'adresse URL du serveur	Temps de connexion au serveur (msec)	Vitesse de transfert (kbit/sec)	Temps de téléchargement (msec)	Taille de fichiers téléchargés (octets)	Vitesse de transfert arrondie (kbit/sec)	Bande passante du réseau
www.google.fr	922	9.8	311	3145	10	**9,6 kbit/sec**
www.google.com	1021	7.3	20	151	8	
www.orange.com	991	8.6	110	973	9	
www.esiee.fr	1161	8.3	30	258	9	
www.int-evry.fr/banner1.jpg	511	6.7	40	277	7	
www.google.fr	962	1.1	614	3145	2	**28,8 kbit/sec**
www.google.com	1022	1.6	91	151	2	
www.orange.com	661	1.2	741	973	2	
www.esiee.fr	1171	1.3	180	258	2	
www.int-evry.fr/banner1.jpg	531	1.5	170	277	2	
www.google.fr	892	3.3	911	3145	4	**112 kbit/sec**
www.google.com	991	7.3	20	151	8	
www.orange.com	475	3.8	250	973	4	
www.esiee.fr	1352	8.3	30	258	9	

Annexe 3 Résultats de l'évaluation du streaming audio-vidéo

Date	Heure	Type format audio	Nombre de trames audio	Fréquence d'échantillonnage audio	Débit binaire audio	Taille de la trame	Format géométrique	Débit binaire vidéo	Nombre de I	Nombre de P	Nombre de B	Nombre de trames vidéo	Vitesse de transfert (kbits/sec)	Nombre de trames transférées/sec
03-déc-03	14:20:28	II	64	44	241	160 x 120	SQUARE	30	19	74	186	282	107	10
03-déc-03	14:24:12	II	48	44	272	166 x 100	SQUARE	23.976	29	81	213	323	124	8
03-déc-03	14:28:56	II	48	44	272	166 x 100	SQUARE	23.976	27	81	215	323	120	8
03-déc-03	14:35:35	II	48	44	272	166 x 100	SQUARE	23.976	27	81	215	323	131	9
03-déc-03	14:38:26	II	48	44	272	166 x 100	SQUARE	23.976	27	81	216	324	146	10
03-déc-03	14:42:02	III	112	44	360	240 x 180	SQUARE	30	6	38	43	87	119	4
03-déc-03	14:48:10	III	128	48	89	176 x 120	SQUARE	29.97	52	205	514	776	120	27

Annexe 4 Publications

A. RADU **« Qualité de Service, Contrat de Service dans les Mobiles de Troisième Génération »**, ***International Conference of Telecommunication Systems***, Université « Politehnica » de Bucarest, mars 2002

A. RADU , JM SAHUT **« INTELLIGENT AGENTS FOR QUALITY OF SERVICE EVALUATION IN MULTIMEDIA SERVICES »** ***Wireless Information System 2003*** , Avril 2003, Angers, France

A. RADU, G.BAUDOIN **« Evaluation in multimedia services »** ***First International Working Conference on Performance Modelling And Evaluation Of Heteregenous Networks (HET-NETs '03)***, London, UK, June 2003

A. RADU, G.BAUDOIN **« Quality of Service evaluation in Multimedia services »** 2003 ***International Conference on Software, Telecommunications and Computer Networks***, October 07-10, 2003, Split, Dubrovnik (Croatia)

A. RADU, G.BAUDOIN **« INTELLIGENT AGENTS FOR QUALITY OF SERVICE EVALUATION IN MULTIMEDIA SERVICES »**, ***IADIS International WWW/Internet 2003***, Algarve,Portugal, 5-8 November 2003

Annexe 5 Classes JAVA des Agents

Dans ce paragrphe nous avons présenté quelques exemple de type d'Agent (L'agent Manager, L'Agent Ping et l'Agent Download). D'autre Agents (Mail ou Vidéo) sont disponibles, mais pour un raison de la taille du code des programmes, ceux-ci peuvent être présentés dans ces annexes.

Agent Manager

FormSubclass.java

```
import javax.microedition.lcdui.*;
 public class FormSubclass extends Form implements CommandListener
{
private TestAgent myAgent;
private Command cmdBack;
private Command pressMe;
private TestGuiMIDP midlet;
 StringItem usrn;
StringItem hote;
 //StringItem tRep;
//public StringItem min = new StringItem("Min:", "");
//public StringItem max = new StringItem("Max:", "");
//public StringItem med = new StringItem("Méd:", "");
public StringItem connTime = new StringItem("Temps de connexion:", "");
public StringItem vitesse = new StringItem("Vitesse < ", "");
public StringItem mail = new StringItem("Mail Info: ", "");
public StringItem beg = new StringItem("To begin application press Start button:", "");
public FormSubclass(String title, TestGuiMIDP midlet)
{
super(title);
this.midlet = midlet;
//Crating elements on the form
  //StringItem
  usrn = new StringItem("UserName:", midlet.x.getString());
  //StringItem
  hote = new StringItem("Hote:", midlet.y.getString());
  //StringItem
```

```
  //tRep = new StringItem("Temps de repause:", midlet.y.getString()+ midlet.lista.getString(midlet.lista.getSelectedIndex()));
connTime.setText(midlet.connTime+" ms");
vitesse.setText(midlet.vitesse+" Ko/s");
mail.setText(midlet.mail);
 append(usrn);
append(hote);
//append(tRep);
//append(min);
//append(max);
//append(med);
 append(connTime);
append(vitesse);
append(mail);
// Commands
cmdBack = new Command("Précéd", Command.BACK, 1);
pressMe = new Command("Démarr", Command.SCREEN, 2);
// Add command and listen for events
addCommand(cmdBack);
setCommandListener(this);
addCommand(pressMe);
setCommandListener(this);
System.out.println(this.midlet.x.getString());

 }
public void commandAction(Command c, Displayable s)
{
if (c == cmdBack)
midlet.displayMainMIDlet();
else if (c == pressMe)
   System.out.println(" sexy MF!");
   //hos.setText(midlet.y.getString());//=MainMIDlet.midlet.y.getString(this);
```

TestAgent.java

```
import jade.core.*;
import jade.core.behaviours.*;
import jade.lang.acl.*;



public class TestAgent extends Agent {
private AID     dest;
private TestGui gui;
 public void setup() {
// get the parameter (i.e. the target agent's local name)
```

```
        try {
           dest = new AID((String)getArguments()[0],AID.ISLOCALNAME);
        }
        catch(Exception e) {
           System.out.println("WARNING: no target agent");
        }
// create the GUI
        //gui = new TestGuiMIDP();
try {
           gui = (TestGui)Class.forName("TestGuiMIDP").newInstance();
        }
        catch (Exception e) {
           try {
                        System.out.println(e);
              gui = (TestGui)Class.forName("examples.leap.testagent.TestGuiAWT").newInstance();
           }
           catch (Exception ex) {
              System.out.println("No GUI for this Agent");
           }
        }
        System.out.println("inainte de gui initialise");
        gui.initialise(this);
        addBehaviour(new RBehaviour(this));
     }
public void setDest(String s){
        try {
           dest = new AID(s,AID.ISLOCALNAME);
        }
        catch(Exception e) {
           System.out.println("WARNING: no target agent");
        }
void sendMessage(String s) {
        ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
        msg.setContent(s);
        msg.addReceiver(dest);
        send(msg);
     }
class RBehaviour extends SimpleBehaviour {
public RBehaviour(Agent a) {
           super(a);
        }
public void action() {
           ACLMessage m = myAgent.blockingReceive();
           if (m != null)
              gui.msgRecv(m.getContent());
```

```
    }
  public boolean done() {
        return false;
    }
```

TestGui.java

```
public interface TestGui {
void initialise(TestAgent agent);
  void msgRecv(String s);
   }
```

TestGUIMIDP.java

```
import javax.microedition.lcdui.*;
import javax.microedition.midlet.*;
public class TestGuiMIDP extends MIDlet implements TestGui, CommandListener {
 private TestAgent myAgent;
  private Display display; // Our display
   private Form frmMain; // Main form
   private FormSubclass frmSub; // Subclass of form
   private Command cmdExit; // Command to exit
   private Command cmdStart; // Command to call form subclass
   public TextField x = new TextField("", "username", 200, TextField.ANY );
   public TextField y = new TextField("", "www", 200, TextField.ANY );
   public StringItem usr = new StringItem("UserName", "");
   public String vitesse;
   public String connTime;
   public String mail;
 public TestGuiMIDP() {
      System.out.println("Sunt in constructor");
      System.out.println(this);
      display = Display.getDisplay(this);
      frmMain = new Form("Web Connection");
      frmMain.append(usr);
      frmMain.append(x);
      frmMain.append(new StringItem(" Host Name ", ""));
      frmMain.append(y);
      cmdExit = new Command("Quitter", Command.EXIT, 1);
      cmdStart = new Command("Ok", Command.SCREEN, 2);
      frmMain.addCommand(cmdExit);
      frmMain.addCommand(cmdStart);
 frmMain.setCommandListener(this); }
 public void initialise(TestAgent agent) {
```

```
        System.out.println("in initialise");
        myAgent = agent;
        displayMainMIDlet();
    }
public void startApp ()
    {
//    displayMainMIDlet();
    }
public void destroyApp (boolean unconditional)
    {
    }public void pauseApp ()
    public void displayMainMIDlet()
    {
    System.out.println("Ajung in displayMainMIDlet");
    display.getDisplay(jade.Boot.midlet).setCurrent(frmMain);
    }
    public void commandAction(Command c, Displayable d)
    {
        if (c == cmdExit)
        {
            destroyApp(false);
            notifyDestroyed();
        }
        else if (c == cmdStart)
        {
            myAgent.setDest("P");
            if (y.getString().indexOf("http://")<0) myAgent.sendMessage("START"+"http://"+y.getString());
            else myAgent.sendMessage("START"+y.getString());
           //frmSub = new FormSubclass("Subclass of Form", this);
            //System.out.println(y.getString());
            //display.getDisplay(jade.Boot.midlet).setCurrent(frmSub); }
    }
public void msgRecv(String s) {
System.out.println(s);
//        r.setText(""+(++rCnt));
//        Display.getDisplay(jade.Boot.midlet).setCurrent(this);
    System.out.println("Acum iar primesc mesaj"+s);
    if (s.indexOf("P")>=0){
        frmMain.setTitle("Pinging...");
        s=s.substring(2);
         connTime=s;
         myAgent.setDest("D");
         if (y.getString().indexOf("http://")<0) myAgent.sendMessage("START"+"http://"+y.getString());
         else myAgent.sendMessage("START"+y.getString());
        }
```

```
   if (s.indexOf("D")>=0){
      frmMain.setTitle("Downloading...");
      s=s.substring(2);
      String writerMessage=new String(s);
      s=s.substring(0,s.indexOf(" "));
      vitesse=s;
      //s=s.substring(s.indexOf("/"));
      frmMain.setTitle("Writing...");
      myAgent.setDest("W");
      myAgent.sendMessage("WRITE"+x.getString()+"web"+" "+y.getString()+" "+connTime+" "+writerMessage);
      myAgent.setDest("Mail");
      myAgent.sendMessage("START"+x.getString());
}
    if (s.indexOf("Mail")>=0){
      mail=s.substring(5);
      myAgent.setDest("W");
      myAgent.sendMessage("WRITE"+x.getString()+"mail"+" "+s);
      frmSub = new FormSubclass("Results", this);
      display.getDisplay(jade.Boot.midlet).setCurrent(frmSub);
      }

      }

}
```

Agent PING

pAg.java

```
package ping;
 import jade.core.*;
import jade.core.behaviours.*;
import jade.lang.acl.*;
import java.lang.*;
import java.io.*;
import java.util.*;
import javax.microedition.io.*;
 public class pAg extends Agent {
private AID     dest;
    public class ping{
    public void ping(){
        }
public long run(String url){
         try {
            long start=System.currentTimeMillis();
            HttpConnection hc = (HttpConnection)Connector.open(url, Connector.READ );
            DataInputStream dis = hc.openDataInputStream();
            dis.close();
            hc.close();
            long end=System.currentTimeMillis();
            System.out.println(start);
            System.out.println(end);
            return end-start;
         } catch (Exception e) {
                return -1;
           }
    }
  }//end class ping
public void setup() {
// get the parameter (i.e. the target agent's local name)
        try {
            dest = new AID((String)getArguments()[0],AID.ISLOCALNAME);
        }
        catch(Exception e) {
            System.out.println("WARNING: no target agent");
        }
        addBehaviour(new RBehaviour(this));
```

```
    }
    void sendMessage(String s) {
       System.out.println("P"+s);
       ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
            msg.setContent(s);
       msg.addReceiver(dest);
       send(msg);
    }
    void msgRecv(String s){//received mesage is always START
    System.out.println("start ping");
    StringBuffer st = new StringBuffer();
    ping p=new ping();
    st.append(p.run(s.substring(5)));
    sendMessage("P "+st.toString());
    System.out.println("end ping");
    }
    class RBehaviour extends SimpleBehaviour {
public RBehaviour(Agent a) {
          super(a);
       }
       public void action() {
          ACLMessage m = myAgent.blockingReceive();
          if (m != null)
             msgRecv(m.getContent());
       }
       public boolean done() {
```

Agent DOWNLOAD

dAg.java

```
package down;
 import jade.core.*;
import jade.core.behaviours.*;
import jade.lang.acl.*;
import java.lang.*;
import java.io.*;
import java.util.*;
import javax.microedition.io.*;
 public class dAg extends Agent {
   private AID     dest; public class down{
   private long   bytes=0;

   public void down(){
      }
   public void clearBytes(){
      bytes=0;
      }
   public void chBytes(){
      //System.out.println(bytes);
      bytes=bytes+1;
      }
   public long getBytes(){
      return bytes;
      }
   public long run(String url){
       try {
          HttpConnection hc = (HttpConnection)Connector.open(url, Connector.READ );
          DataInputStream dis = hc.openDataInputStream();
          System.out.println( "Host:             " + hc.getHost() );
          System.out.println( "Protocol:         " + hc.getProtocol() );
          int ch;
            // Check the Content-Length first
            long start=System.currentTimeMillis();
            long len = hc.getLength();
            System.out.println("Lungimea:" + len);
            if (len!=-1) {
               //bytes=len;
               for (int i = 0;i<len;i++)
                  if ((ch = dis.read())!= -1) {
```

```
                    chBytes();
                }
            } else {
                // if the content-length is not available
                while ((ch = dis.read()) != -1)
                    chBytes();
            }
            System.out.println( "Sfarsit" );
            dis.close();
            hc.close();
            long end=System.currentTimeMillis();
            return end-start;
        } catch (Exception e) {return -1;}
    }
}// end class down
    public void setup() {
        // get the parameter (i.e. the target agent's local name)
        try {
            dest = new AID((String)getArguments()[0],AID.ISLOCALNAME);
        }
        catch(Exception e) {
            System.out.println("WARNING: no target agent");
        }
        addBehaviour(new RBehaviour(this));
    }
    void sendMessage(String s) {
        ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
                msg.setContent(s);
        msg.addReceiver(dest);
        send(msg);
    }
void msgRecv(String s){
    System.out.println("D"+s);
    down d=new down();
    StringBuffer st= new StringBuffer();
    long elapsed=d.run(s.substring(5));
    st.append(d.getBytes()/elapsed+1);
    st.append(" ");
    st.append(elapsed);
    st.append(" ");
    st.append(d.getBytes());
    d.clearBytes();
    sendMessage("D "+st.toString());
    }
class RBehaviour extends SimpleBehaviour {
```

```
public RBehaviour(Agent a) {
        super(a);
    }
public void action() {
        ACLMessage m = myAgent.blockingReceive();
        if (m != null)
           msgRecv(m.getContent());
    }
public boolean done() {
        return false;
    }
} // End of internal RBehaviour class
}
```